Based on National Curriculum of Pakistan 20

Textbook of Computer Science Grade 12

NATIONAL BOOK FOUNDATION
AS
FEDERAL TEXTBOOK BOARD, ISLAMABAD

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Based on National Curriculum of Pakistan 2022-23

Textbook of

Computer Science

Grade 12

National Curriculum Council
Ministry of Federal Education and Professional Training

National Book Foundation
as
Federal Textbook Board
Islamabad

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

The Practical Importance of the Textbook in Everyday Life

The following is a unit-by-unit summary of what is presented in each unit with its practical importance in our everyday life.

UNIT 1 Computer Systems: This unit focuses on Human Computer Interaction (HCI). HCI is a field of study about designing and development of interactive systems that are efficient, effective and user-friendly while using digital devices. HCI plays an important role in interaction between the computer system and the user in providing ways through which the user can input data, receive feedback and navigate through the system.

UNIT 2 Computational Thinking & Algorithms: This unit is about developing algorithms which is the first step for writing computer programs. Learning how to develop algorithms holds immense significance for students as it enhances their ability to think critically and provide structured approach to computational problem-solving

UNIT 3 Programming Fundamentals: In this unit students learn programming fundamentals in Python. Python is a high level object-oriented programming language. This unit describes advanced programming constructs such as data structures, file handling and databases and how to implement complex algorithms in Python. In today's modern world, having computer programming skill is essential as it plays an important role in automation of tasks that we perform in our daily life.

UNIT 4 Data and Analysis: This unit focuses on data analysis. It contains material on Machine Learning (ML). ML is a powerful technology that is transforming many industries by making processes efficient and effective. It provides solutions to many problems related to various fields such as marketing and advertising, financial services, healthcare and self-driving cars. Material on data visualization is also explored which is representation of information in the form of chart, graph, maps, diagram, etc.

UNIT 5 Applications of Computer Science: This unit describes the applications of Internet of Things (IoT), Blockchain and Cloud Computing that are applicable in government and private sectors in Pakistan. Applications of these enhances our daily life by improving productivity, efficiency and security. Material on neural networks and deep learning is also presented that is used for development of complex systems. Data sharing and privacy conflicts are explored at the end of the unit to understand issues related with it.

UNIT 6 Impacts of Computing: This unit focuses on issues faced by people when collaborating on digital or online platforms. Today, much of our daily life revolves around the Internet and this exposes us to a wide range of online threats. This unit describes security measures and safe practices such as 2FA, biometric verification and secure ways for transmitting data, to mitigate online threats.

UNIT 7 Digital Literacy: This unit presents material that describes the creation of artefact that answers a research question, communicates results and conclusions through digital resources or tools. It plays an important role in strengthening research, analytical skills and fosters digital literacy.

UNIT 8 Entrepreneurship in Digital Age: This unit explores the creation of a Minimum Viable Product (MVP) for entrepreneurs and businesses. MVP provides low-risk testing ground before making huge investment into a product. It allows entrepreneurs and businesses to validate their business ideas, minimize costs and mitigate risks while launching a new product.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Table of Contents

Unit 1: Computer Systems

1.1 System/Device Usability, Security, and Accessibility..... 08

1.2 Human-Computer Interaction (HCI) 22

1.3 Trade-offs between Usability and Security in Computing Systems..... 30

Unit 2: Computational Thinking & Algorithms

2.1 Data Structures.....41

2.2 Evaluating Computational Solutions.....49

Unit 3: Programming Fundamentals

3.1 The Programming Paradigm.....60

3.2 Programming Constructs in Python.....63

3.3 Techniques for Testing & Debugging.....90

Unit 4: Data and Analysis

4.1 Data Types.....101

4.2 Data Visualization.....108

4.3 Hypothesis Formulation and Hypothesis Testing.....112

Unit 5: Applications of Computer Science

5.1 Internet of Things (IoT).....127

5.2 Designing IoT-based Applications..... 127

5.3 IoT Applications Applicable to Pakistan..... 129

5.4 Blockchain..... 137

5.5 Blockchain Applications Applicable to Pakistan..... 137

5.6 Cloud Computing.....139

5.7 Neural Networks and Deep Learning.....141

5.8 Data Sharing and Privacy..... 148

Unit 6: Impacts of Computing

6.1 Security Protocols.....160

6.2 Risks of Sharing Private Information..... 163

6.3 Best Practices to Prevent Identity Theft.....163

6.4 Cyber-attacks..... 164

6.5 Security Methods.....165

Downloaded from Atta Shad College Digital Library

Scanned by CamScanner

6.6 Safe Transmission of Data..... 166
6.7 Security Protocols.....168
6.8 Troubleshoot Security Problems.....169
6.9 Identifying a Cybersecurity Threat..... 170
6.10 Computational Perspectives.....171
6.11 Computing Applications..... 172
6.12 Resources for Equal Information Accessibility..... 173
6.13 Collaborative Tools.....174

Unit 7: Digital Literacy

7.1 Performing Advanced Searches to Locate Information.....184

Unit 8: Entrepreneurship in Digital Age

8.1 Building and Launching the MVP (Minimum Viable Product)... 197
8.2 Iterative Development..... 202
8.3 Conclusion and Future Directions.....206

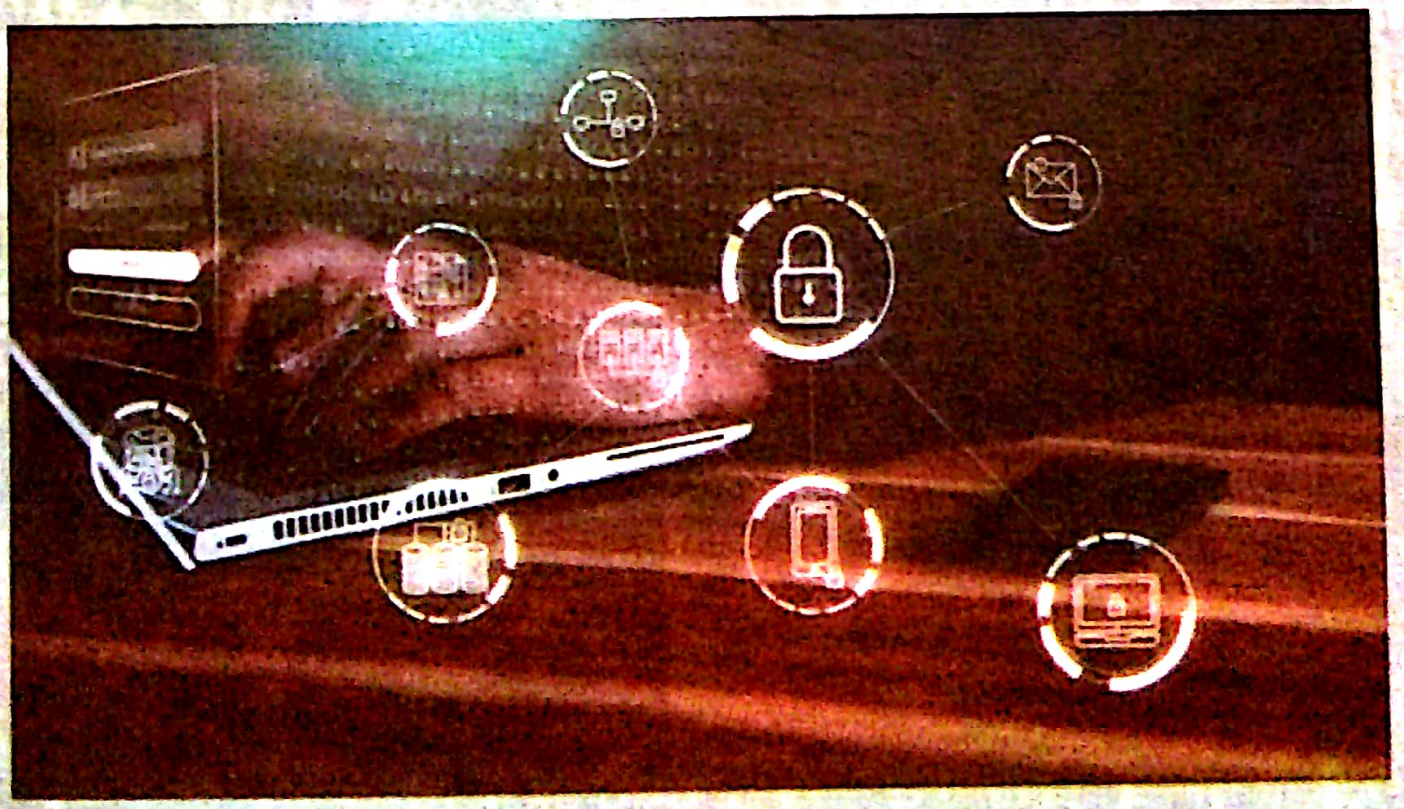
Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

1 Computer Systems

Learning Outcomes

At the end of this unit students will be able to:

- explain the usability, security and accessibility of devices, the systems they are integrated with.
- explain human interaction with computer systems in terms of:
  - Usability
  - Common problems
  - Methods for improvements
  - Ethical, social, economic, and environmental implications
- identify and explain tradeoffs between the usability and security of computing systems, recommend cybersecurity measures by considering different factors such as efficiency, cost, privacy, and ethics.



Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Introduction

In today's rapidly evolving digital world, technology plays a central role in our everyday lives. Whether we are using smartphones, accessing online services, or interacting with smart devices at home, the design and functionality of these technologies deeply impact our experiences. As we dive deeper into the world of technology, three key concepts emerge as critical to our interaction with these systems. These are **usability, security, and accessibility.**

- **Usability** ensures that devices and systems are easy to use and understand. Without it, even advanced technology can become frustrating and ineffective.
- **Security** protects our data and interactions in a world of evolving cyber threats. Strong measures are essential to safeguard sensitive information and prevent severe consequences.
- **Accessibility** guarantees that technology is available to everyone, including those with disabilities. As our world becomes increasingly digital, it is important that no one is left behind. Accessible design makes sure that all users, regardless of their physical abilities, can fully interact with and benefit from technology.

What will You Learn?

This unit explores the key principles of usability, security, and accessibility, highlighting their role in designing effective and inclusive technology. You will also learn about **Human-Computer Interaction (HCI)** and its focus on user needs. Additionally, the unit examines the **trade-offs between usability and security**, providing insights into how designers and engineers balance these often conflicting requirements to create systems that protect users without compromising their experience.

1.1 System/Device Usability, Security, and Accessibility

When devices and systems are designed, they should be easy to use (usability), safe and secure from threats (security), and accessible to everyone, including people with disabilities (accessibility). Understanding tnese aspects is crucial because they affect how we interact with technology and how technology affects our lives.

1.1.1 Device/System Usability

Usability is a key factor in the success of any device or system. It refers to how easily and efficiently users can interact with technology, directly affecting their satisfaction and productivity. By applying usability principles, designers can create user-friendly systems that meet user needs and provide a seamless experience.

Fig.1.1: Human-Computer Interaction (HCI)

A well-designed system with high usability is easy to learn, simple to use, and aligned with user expectations. It enables users to achieve their

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

goals effectively, efficiently, and with minimal effort, ensuring a positive and satisfying experience. Fig.1.1 illustrates such a system.

Key Aspects of Usability:

The following are the key aspects of Usability.

- **Effectiveness:** The degree to which users can successfully complete tasks using the device or system.
- **Efficiency:** The speed and minimal effort with which tasks can be accomplished.
- **Satisfaction:** The user's sense of comfort and positive experience while interacting with the system.

Principles of Usability

Several key principles guide the design of user-friendly systems, ensuring they meet the needs and expectations of their users. These principles include:

- Simplicity
- Consistency
- Feedback
- Error Prevention
- Learnability
- Accessibility

Fig.1.2: Simplicity

1. Simplicity:

The design should be as straightforward as possible, removing unnecessary complexity and focusing on core functionalities.

Example: The Apple iPod (Fig.1.2), with its simple click-wheel interface, allowed users to navigate music/video quickly and easily, without the clutter of extraneous buttons.

Google's homepage (Fig.1.3) is iconic for its simplicity, concentrating on the primary function—searching—without overwhelming users with excessive information or options.

Fig.1.3: Google homepage

2. Consistency:

Interfaces should maintain a consistent design across different parts of the system, making it easier for users to predict how the system will behave and reducing the cognitive load required to use it.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Example: Microsoft Office uses similar menus and toolbars across Word, Excel, and PowerPoint, helping users transition smoothly between applications, as shown in Fig.1.4.

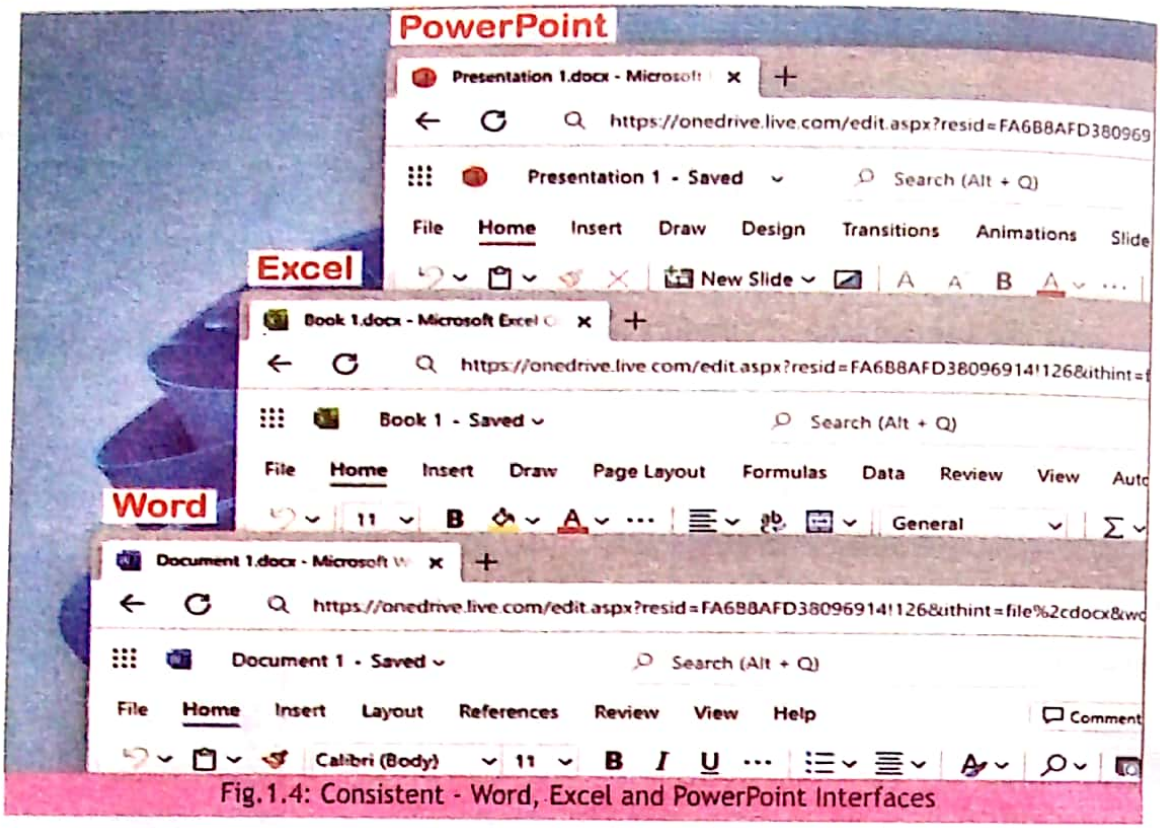


Fig.1.4: Consistent - Word, Excel and PowerPoint Interfaces

3. Feedback:

The system should provide clear, immediate feedback to users regarding their actions, helping them understand if their inputs were successful and what the system is doing in response.

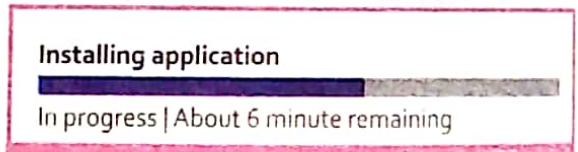


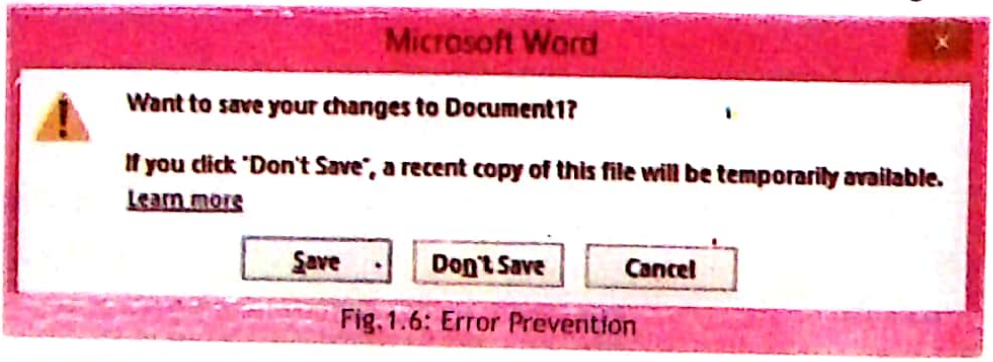
Fig.1.5: Application installation progress bar - feedback

Example: E-commerce sites like Amazon provide immediate feedback when an item is added to the shopping cart, often with a pop-up or visual confirmation, reassuring the user that their action was successful. Another example is the progress bar in software installations, as shown in Fig.1.5, which indicates the ongoing process and how much time is remaining.

4. Error Prevention:

The design should prevent errors before they occur and offer easy ways to correct them if they do.

Example: When you try to close a document without saving, a word processor like Microsoft Word prompts you to save your work, preventing accidental data loss. As shown in Fig.1.6.


Microsoft Word
Want to save your changes to Document1?
If you click "Don't Save", a recent copy of this file will be temporarily available.
Learn more
Save
Don't Save
Cancel
Fig.1.6: Error Prevention

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5. Learnability:

The system should be easy for new users to learn, allowing them to become proficient quickly, with minimal training or guidance.

Example: Mobile apps often include a tutorial or walkthrough for first-time users, helping them get started quickly.

6. Accessibility:

The system should be designed so that it can be easily used by everyone, regardless of their abilities. This includes making sure that people with disabilities can also use it comfortably. The goal is to ensure that the system is inclusive, meaning it works well for as many people as possible, no matter their physical or cognitive abilities.

Example1: Voice Over feature on Phones (Fig.1.7)

Voice Over feature allows visually impaired users to interact with their Phones by reading out text on the screen. Users can navigate through apps, read messages, and even use the camera with voice guidance, making the device accessible to those who are blind or have low vision.

Fig.1.7: Voice Over feature on Phones

Example2: TV Remote Controls (Device Accessibility)

Some TV remotes are designed with large buttons and high-contrast colors to assist users with vision or dexterity issues. These remotes are easier to see and press, making them more accessible to elderly users, as shown in Fig.1.8.

Fig.1.8: TV Remote Controls

1.1.2 Usability Testing

Usability testing is a key method used to evaluate how user-friendly and effective a system or device is. There are several types of usability testing, each with a specific focus and approach.

The following are some common types of usability testing methods.

1. Moderated Usability Testing

In this type of testing, a facilitator or moderator is present to guide users through tasks, ask questions, and observe behaviors in real-time. The moderator can also interact with users to clarify instructions or probe deeper into specific issues. Fig. 1.9 shows the whole process.

Example: A company developing a new software application conducts moderated usability testing by inviting participants to a lab environment. The moderator asks users to perform specific tasks, such as navigating through the app's features or completing a transaction. The moderator observes how easily users can accomplish these tasks and asks questions about their experience, helping to identify any usability issues.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

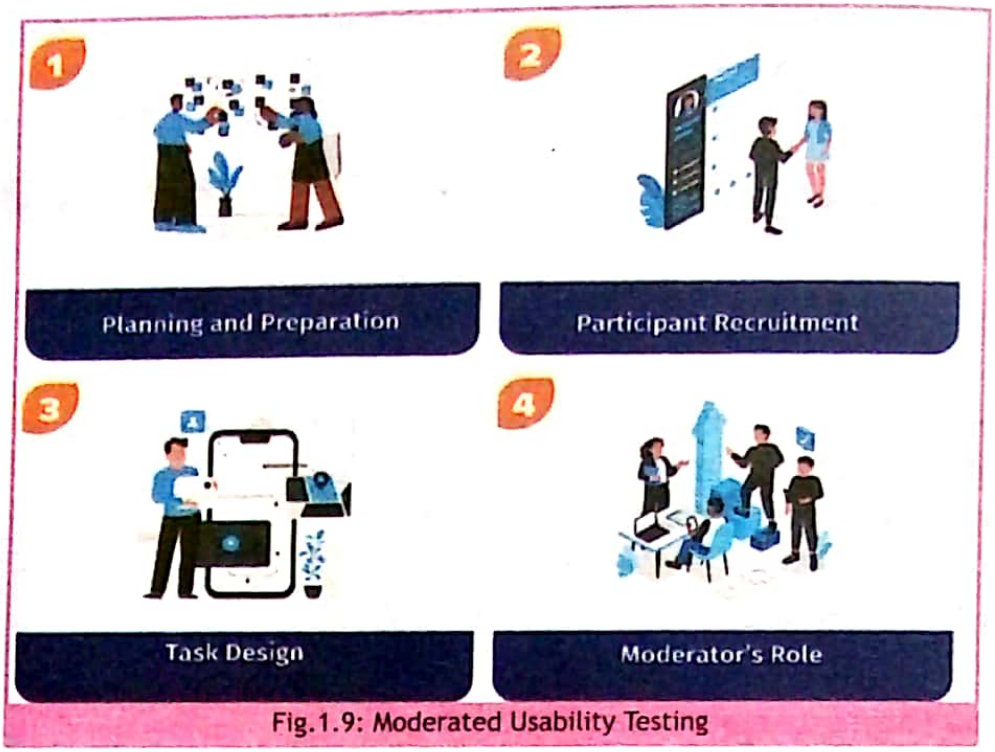


Fig.1.9: Moderated Usability Testing

2. Unmoderated Usability Testing

In unmoderated usability testing, users complete tasks on their own without a moderator's direct involvement. This type of testing is often conducted remotely, allowing participants to use the system or device in their natural environment. The sessions are typically recorded for later analysis, as shown in Fig.1.10.

Fig.1.10: Unmoderated Usability Testing

Example: An e-commerce website wants to test how easily customers can find and purchase products. The company conducts unmoderated usability testing by sending out tasks for users to complete at home, such as searching for a product, adding it to the cart, and completing the checkout process. The user interactions are recorded, and the company reviews the footage to identify any pain points or areas of confusion.

3. Qualitative Usability Testing

This approach focuses on understanding the user's experience, thoughts, and feelings while interacting with the system or device. The goal is to gather in-depth insights into user behavior, including what they find intuitive or challenging. As shown in Fig.1.11.

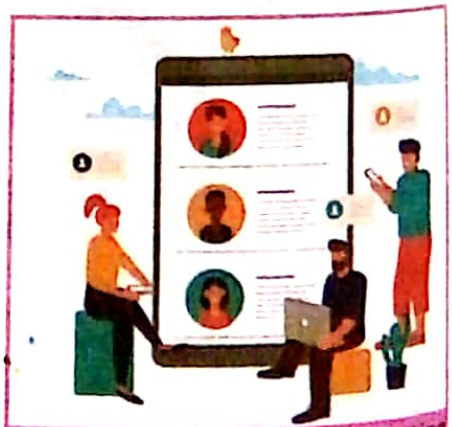


Fig.1.11: Qualitative Usability Testing

Example: A mobile app developer conducts qualitative usability testing by interviewing users after they use the app. Users are asked open-ended questions about their experience, such as what they liked or disliked and any challenges they encountered. This feedback helps the developer understand the user's perspective and make improvements to the app's design.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

4. Quantitative Usability Testing

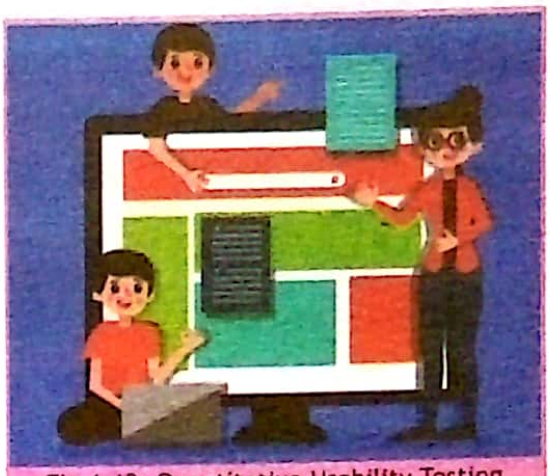


Fig.1.12: Quantitative Usability Testing

Quantitative usability testing collects numerical data to measure usability aspects like task completion time, error rates, and success rates. This method is useful for comparing different versions of a system or device or setting benchmarks for usability.

Example: A company tests two versions of a website's navigation menu. In a quantitative usability test, users are timed as they attempt to find specific information using each menu version. The company collects data on how long it takes users to complete the tasks and how often they make errors. This data is then analyzed to determine which menu is more efficient and user-friendly. As shown in Fig.1.12.

5. Remote Usability Testing

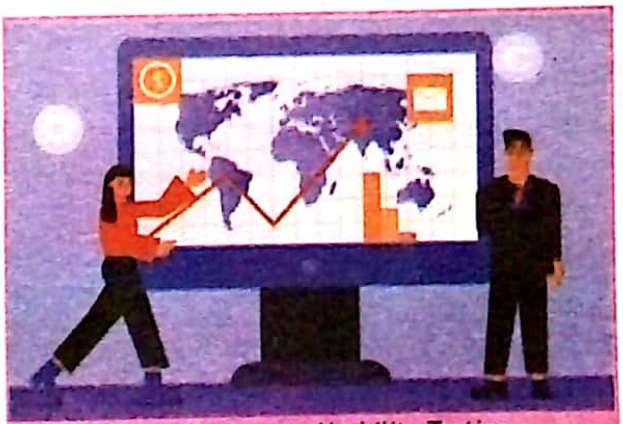


Fig.1.13: Remote Usability Testing

Remote usability testing allows users to test a system or device from their own location, using their own devices. This type of testing can be either moderated or unmoderated, and it is particularly useful for reaching a wider audience or observing how users interact with a product in their natural environment.

Example: A global software company wants to test a new feature with users in different countries. Remote usability testing is conducted, with participants using the software on their own computers while being observed through screen-sharing technology. The company gains insights into how the feature performs in different regions and environments without needing to bring users into a lab. As shown in Fig.1.13.

6. A/B Testing

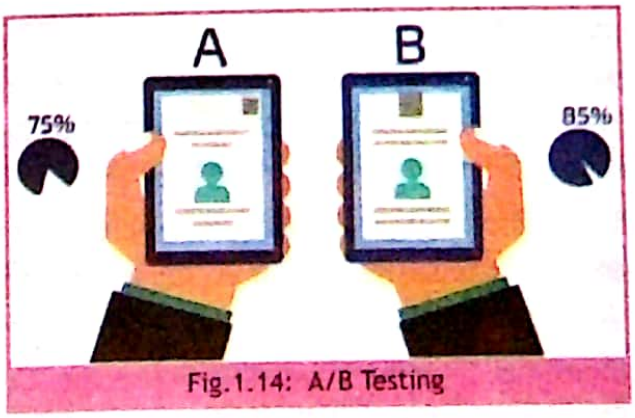


Fig.1.14: A/B Testing

A/B testing, also known as split testing, involves comparing two versions of a system or device to see which one performs better in terms of usability, as shown in Fig.1.14. This type of testing is commonly used in web design and online services.

Example: An online retailer wants to determine whether a new homepage layout will lead to more sales. They create two versions of the homepage and use A/B testing to randomly show each version to different users. By comparing the sales metrics and user engagement data between the two versions, the retailer can determine which layout is more effective.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

1.1.3 Common Usability Issues and How to Avoid Them

Usability issues can significantly impact the effectiveness and user satisfaction of a system or device. When users encounter problems with usability, they may become frustrated, make errors, or even abandon the product altogether. Understanding common usability issues and knowing how to avoid them is crucial in designing user-friendly systems.

The following are some common usability issues, along with practical solutions and examples.

1. Complex Interfaces

Complex interfaces are those that overwhelm users with too much information, too many options, or overly complicated layouts. When users are faced with an interface that is difficult to understand or navigate, they are more likely to make mistakes, take longer to complete tasks, or feel frustrated.

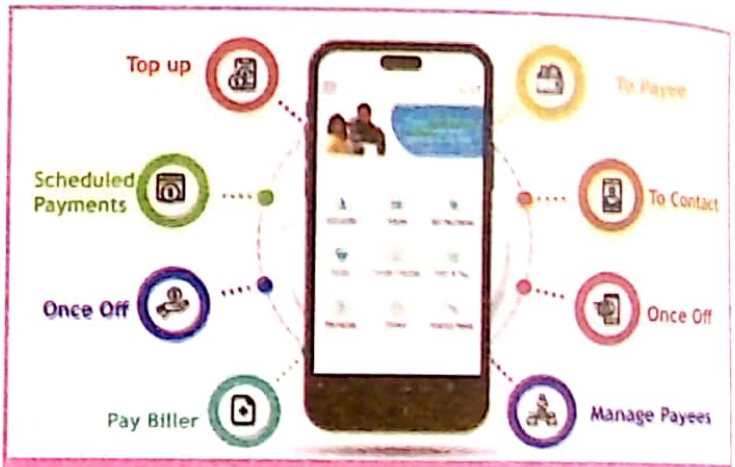


Fig.1.15: Complex Interfaces

Solution: To avoid complexity, focus on simplicity. Simplify the interface by prioritizing core functions and removing unnecessary features. The design should highlight the most important tasks and provide a clear and intuitive path for users to accomplish them.

Example: Example: Consider a mobile banking app (Fig.1.15) that initially has dozens of features crammed into a single screen. Users might find it difficult to locate basic functions like checking their balance or transferring funds. By simplifying the interface—perhaps by organizing features into easily accessible categories or hiding advanced options behind menus—the app becomes much easier to use. A good example is the early versions of the PayPal app, which went through multiple iterations to simplify its interface, eventually focusing on core actions like sending money and checking balances on the main screen.

2. Inconsistent Navigation

Inconsistent navigation occurs when the navigation structure or elements change from one section of the system to another. This inconsistency can confuse users, making it difficult for them to predict how to move around the system or find the information they need. As a result, users may feel lost or frustrated.

Solution: To maintain consistency, ensure that navigation menus, buttons, and other interface elements remain uniform across all parts of the system. Consistency in design allows users to develop a mental model of how the system works, making it easier for them to navigate.

Example: Imagine a website where the "Contact Us" button is located at the top of the page on some sections and at the bottom on others. Users might struggle to find the contact information depending on where they are on the site. A solution would be to place the "Contact Us" button in the same location on every page, such as in a fixed header or footer. Amazon's website, for

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

instance, keeps its navigation consistent across different pages, with the search bar and navigation menu always in the same place, helping users find what they need quickly.

3. Poor Error Messages

Fig.1.16: Poor Error Messages

Poor error messages are those that are unclear, overly technical, or provide little to no guidance on how to resolve the issue. When users encounter such error messages, they can become confused, frustrated, and unable to continue using the system effectively.

Solution: Write error messages that are clear, concise, and helpful. Good error messages should explain what went wrong in plain language and provide actionable steps that users can take to fix the problem.

Example: Consider an online form where a user enters an incorrect date format, and the system simply displays "Error 150: Invalid input.", as shown in Fig.1.16." This message is vague and doesn't help the user correct the mistake. Instead, the error message should read something like, "Please enter the date in the format MM/DD/YYYY." This clear guidance helps the user understand the issue and correct it immediately. Google's forms often provide such clear and helpful error messages, ensuring users can easily fix any mistakes.

4. Lack of Accessibility

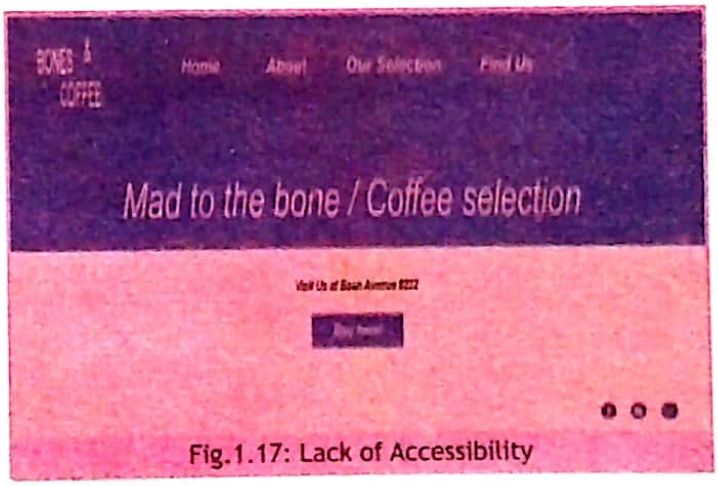


Fig.1.17: Lack of Accessibility

A lack of accessibility means that the system or device is not designed to accommodate users with disabilities, such as those who are visually impaired, hearing impaired, or have motor disabilities. Ignoring accessibility can exclude a significant portion of potential users and lead to a poor user experience for those with disabilities.

Solution: Implement accessibility features to make the system usable for people with a wide range of abilities. This includes adding screen reader support, keyboard navigation, alternative text for images, and options for adjusting text size and color contrast.

Example: Consider a website with small text and low contrast colors that are difficult for visually impaired users to read, as shown in Fig.1.17. By implementing accessibility features like text enlargement options, high-contrast color schemes, and ensuring the website is compatible with screen readers, the website becomes more accessible. Apple's website, for example, offers a high-contrast mode and supports screen readers, making it accessible to users with visual impairments.

1.1.4 Device/System Security

Security in the context of devices and systems refers to the measures and controls implemented to protect information and systems from unauthorized access, use, disclosure, disruption,

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

modification, or destruction. It aims to safeguard data and ensure that systems function as intended. The following are the key objectives of security.

Key Objectives:

Confidentiality: Ensuring that sensitive information is accessible only to authorized users.

Integrity: Protecting data from being altered or tampered with by unauthorized individuals.

Availability: Ensuring that systems and data are available for use when needed.

Principles of Security:

The development and maintenance of secure systems rely on several foundational principles, commonly referred to as the CIA triad—Confidentiality, Integrity, and Availability, as shown in Fig.1.18. These principles serve as the cornerstone for designing systems that protect data and ensure that it remains safe and accessible under various conditions.

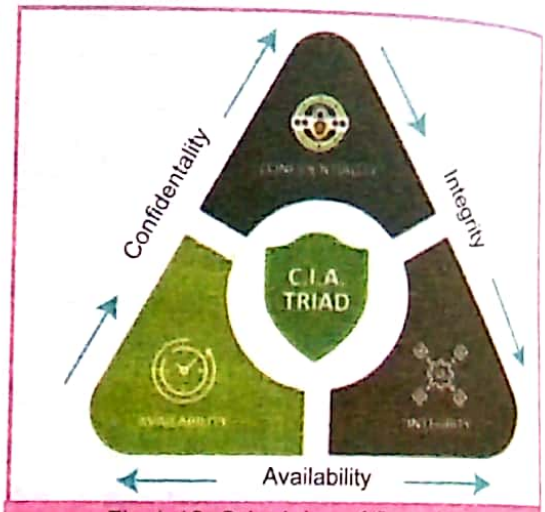


Fig.1.18: Principles of Security

1. Confidentiality

Confidentiality involves protecting sensitive information from unauthorized access, ensuring that only those who have the appropriate permissions or clearance can view or use the data.

Example: Consider a company's database that stores customer credit card information. To keep this data confidential, the company might encrypt the credit card numbers before storing them in the database. This way, even if a hacker gains access to the database, they would not be able to read or misuse the credit card information without the decryption key.

2. Integrity

Integrity refers to the accuracy and consistency of data. It ensures that information remains unaltered during storage, transmission, or processing, except by those who are authorized to do so. This principle protects data from being tampered with or corrupted.

Example: When a user sends an important email, such as a contract, he/she might digitally sign it. A digital signature is a way to ensure that the content of the email has not been altered after it is sent. If the recipient receives the email with a valid signature, they can be confident that the content is exactly as intended and has not been tampered with.

3. Availability

Availability ensures that systems and data are accessible and usable when needed, even during or after a disruption. This principle involves designing systems to withstand and recover from failures or attacks quickly.

Example 1: To guarantee availability, a business might implement more than one servers and data storage. For instance, if one server fails, another can take over immediately, preventing stoppage and ensuring that users can continue to access the system without interruption.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Security Policies and Best Practices:

Implementing strong security policies and best practices is essential for protecting devices and systems.

Data Protection Policies: These are guidelines for handling, storing, and transmitting sensitive information to prevent unauthorized access or disclosure. For example a company policy requiring encryption of all sensitive customer data before transmission.

Regular Security Audits: Audits are conducting thorough assessments of systems and networks to identify vulnerabilities and ensure compliance with security standards. For example an organization performing quarterly security audits to check for any gaps in their defenses.

Employee Training and Awareness: Educating employees on security risks and best practices to prevent human error, which is a common cause of security breaches.

Incident Response Planning: Preparing a plan to quickly and effectively respond to security incidents to minimize damage.

Learning from Security Breaches

Case Study: The Target Data Breach of 2013

Overview

In December 2013, Target Corporation, a major U.S. retail chain, suffered a significant data breach that compromised the personal and financial information of millions of customers.

Key Details

- How It Happened: Hackers accessed Target's network through a third-party vendor, using stolen credentials.
- Data Compromised: Approximately 40 million credit and debit card accounts and personal information of 70 million customers were affected.
- Financial Impact: The breach cost Target an estimated $162 million in legal fees, settlements, and security upgrades.
- Reputation Damage: Target's reputation suffered, leading to a decline in customer trust and sales.
- Regulatory Response: Target paid $18.5 million in a settlement with 47 states and implemented stricter security measures.

Lessons Learned

- Strengthen Cybersecurity: Companies must ensure robust security measures, especially with third-party vendors.
- Employee Training: Regular training on recognizing cyber threats is essential.
- Incident Response Plan: A clear plan for responding to breaches can mitigate damage.
- Regular Audits: Conducting security audits helps identify and fix vulnerabilities.

Assignment

Research two additional data breaches. For each breach, include:

i. Background: Company and context
ii. Date of Breach
iii. Data Compromised
iv. Financial Losses
v. Reasons for the Breach
vi. Remedies Implemented

This exercise will help you understand the importance of cybersecurity in today's digital world.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

1.1.5 Device/System Accessibility

Accessibility refers to the design of devices, systems, and services in a way that they can be used by people with a wide range of abilities and disabilities. It involves making technology usable by everyone, including those with visual, auditory, physical, cognitive, or other types of impairments.

Key Aspects of Accessibility:

The following are the key aspects of Accessibility.

- **Inclusivity:** Ensuring that technology is accessible to the widest possible audience.
- **Adaptability:** Allowing users to customize the way they interact with the system to suit their needs.
- **Compliance:** Adhering to accessibility standards and legal requirements to ensure that products are accessible.

Principles of Accessibility:

Accessibility is a fundamental aspect of design that ensures technology can be used by everyone, including individuals with disabilities. The principles of accessibility guide the creation of systems and interfaces that accommodate a wide range of needs. By adhering to these principles, designers can create products that are inclusive and usable by the broadest possible audience.

1. Perceivability

Perceivability means that all users must be able to perceive the information and user interface components, regardless of their sensory abilities. This principle ensures that content is accessible to users with various sensory impairments, such as those who are blind, deaf, or colorblind.

Examples1: Websites should provide text alternatives **(alt text)** for images. This allows screen readers to describe the content of the images to users who are visually impaired. For example an online store includes alt text for product images so that a blind user can hear a description of the item, such as "Red wool sweater with V-neck and ribbed cuffs," through their screen reader.

Examples2: Captions and Transcripts for Audio/Video: Providing captions for videos and transcripts for audio content ensures that users who are deaf or hard of hearing can access the information. For example a YouTube tutorial video on cooking includes captions so that users who are hearing impaired can follow along with the instructions. As shown in Fig.1.19.

Fig.1.19: Transcripts for Audio/Video

2. Operability

Operability refers to the ability of users to navigate and use the interface, regardless of their physical or motor abilities. This principle focuses on ensuring that all users, including those with disabilities, can interact with the system effectively.

Example1: Keyboard Navigation: Users should be able to navigate through the interface using

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

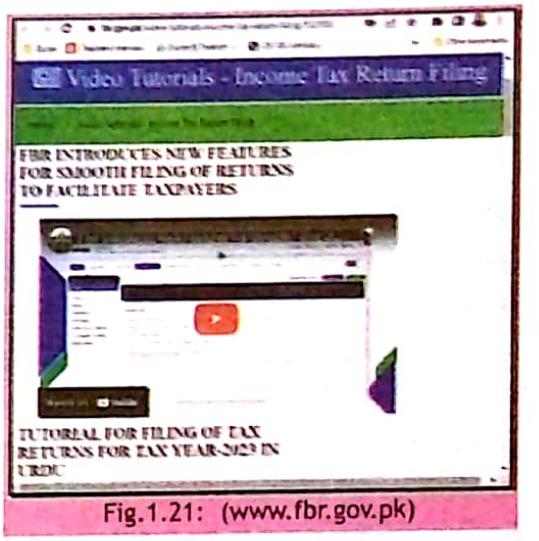
just a keyboard, without needing a mouse.

Fig.1.20: Voice Commands

Example2: Voice Commands: Implementing voice control features allows users to operate systems hands-free. For example a smart home system can be controlled via voice commands, enabling users with mobility impairments to turn on lights or adjust the thermostat without needing to physically interact with the controls. As shown in Fig.1.20.

3. Understandability

Understandability means that the design should make it easy for all users to comprehend the information and know how to operate the interface. This principle focuses on creating content and interactions that are clear and straightforward.


Video Tutorials - Income Tax Return Filing
FBR INTRODUCES NEW FEATURES
FOR SMOOTH FILING OF RETURNS
TO FACILITATE TAXPAYERS
TUTORIAL FOR FILING OF TAX
RETURNS FOR TAX YEAR-2023 IN
URDU
Fig.1.21: (www.fbr.gov.pk)

Example1: Simple Language: A government website (www.fbr.gov.pk) provides instructions for filing taxes in simple language, making the process easier to understand for all tax payers, including those with learning disabilities, as shown in Fig.1.21.

Example2: Error Prevention and Correction: An online form for registering an event highlights required fields and provides real-time feedback if a user forgets to fill out a section, along with instructions on how to correct the mistake.

4. Robustness

Robustness refers to the ability of content to be interpreted by a wide range of devices, including assistive technologies. This principle ensures that content remains accessible even as technologies evolve.

Fig.1.22: Cross-Browser Compatibility

Example1: Cross-Browser Compatibility: Ensuring that a website functions correctly across different web browsers is crucial for accessibility. For example an educational platform tests its website to ensure it works well on Chrome, Firefox, Safari, and Edge(As shown in Fig.1.22), so that students using different browsers have the same experience.

Example2: Support for Multiple Input Devices: A computer device is designed with multiple input options to cater to users with different preferences and needs. For example users can interact with the computer using voice commands for those who find it difficult to use a traditional mouse and keyboard. The touchscreen interface allows for gestures and touch interactions, providing an alternative for users who have difficulty with physical input devices.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

1.1.6 Types of Disabilities and Accessibility Solutions

Designing technology that is inclusive and accessible to individuals with various disabilities involves understanding the unique challenges each type of disability presents and implementing tailored solutions.

The following are some accessibility solutions for different disabilities.

Visual Impairments:

Visual impairments encompass conditions such as blindness, low vision, and color blindness. Accessibility solutions for visual impairments aim to make content perceivable through alternative means.

Solutions and Examples:

Screen Readers: These tools convert text and other visual elements into spoken words or braille. Effective implementation requires semantic HTML and ARIA (Accessible Rich Internet Applications) roles. For example **JAWS** (Job Access with Speech) or **NVDA** (Non Visual Desktop Access) reads aloud content on a web page, such as "Welcome to our website. Today's date is September 1, 2024." This enables blind users to navigate and interact with web content, as shown in Fig.1.23.

Fig.1.23: Screen Readers

Fig.1.24: High Contrast

High Contrast Modes: Enhances readability by using contrasting colors to make text stand out against the background. For example Windows and macOS offer high contrast themes that switch text and background colors to improve visibility. Websites like BBC also provide high contrast modes for users with low vision. (Fig.1.24)

Zoom Functions: Allows users to magnify text and images on the screen to better accommodate low vision. For example Google Chrome and Mozilla Firefox browsers have built-in zoom features that enable users to increase the size of text and images, making them easier to read, as shown in Fig.1.25.

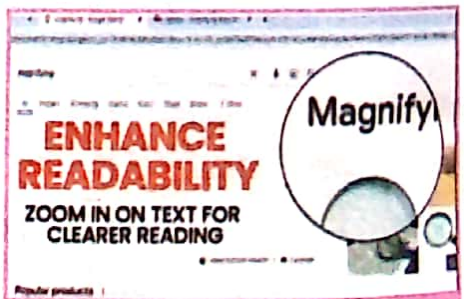


Fig.1.25: Zoom Functions

Hearing Impairments:

Hearing impairments can range from partial hearing loss to complete deafness. Accessibility solutions for hearing impairments focus on providing information through visual and textual means.

Solutions and Examples:

Closed Captions: Text that appears on screen to transcribe spoken dialogue, sound effects, and other audio cues in videos. For example YouTube provides closed captions for videos, enabling users who are deaf or hard of hearing to read the dialogue and sound effects as they watch, as shown in Fig.1.26.

Fig.1.26: Zoom Functions

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Transcripts: Written versions of audio content, which allow users to read the content instead of listening. For example TED Talks provides transcripts for each talk, allowing users to read the content in addition to or instead of watching the video.

Sign Language Interpretation: Includes sign language videos that interpret spoken content into sign language. For example Microsoft Teams and other video conferencing tools allow users to include sign language interpreters in video calls. Similarly, news channels like PTV News enhance accessibility by adding sign language interpreters through a side video, making news broadcasts understandable for deaf viewers, as shown in Fig.1.27.

Fig.1.27: Sign Language Interpretation

Motor Impairments:

Motor impairments affect a person's ability to use traditional input devices like a mouse or keyboard. Accessibility solutions focus on providing alternative methods for interacting with devices.

Solutions and Examples:

Voice Recognition Software: Enables users to control their device and dictate text using voice commands. For example Dragon NaturallySpeaking allows users to navigate their computer, compose documents, and execute commands through voice recognition.

Alternative Input Devices: Includes specialized devices like adaptive keyboards, eye-tracking systems, and adaptive mouse.

On-Screen Keyboards: Virtual keyboards displayed on the screen that can be used with touchscreens or other input methods. For example On-Screen Keyboard provides a virtual keyboard that users can control using a mouse, touch screen, or other adaptive devices, as shown in Fig.1.28.

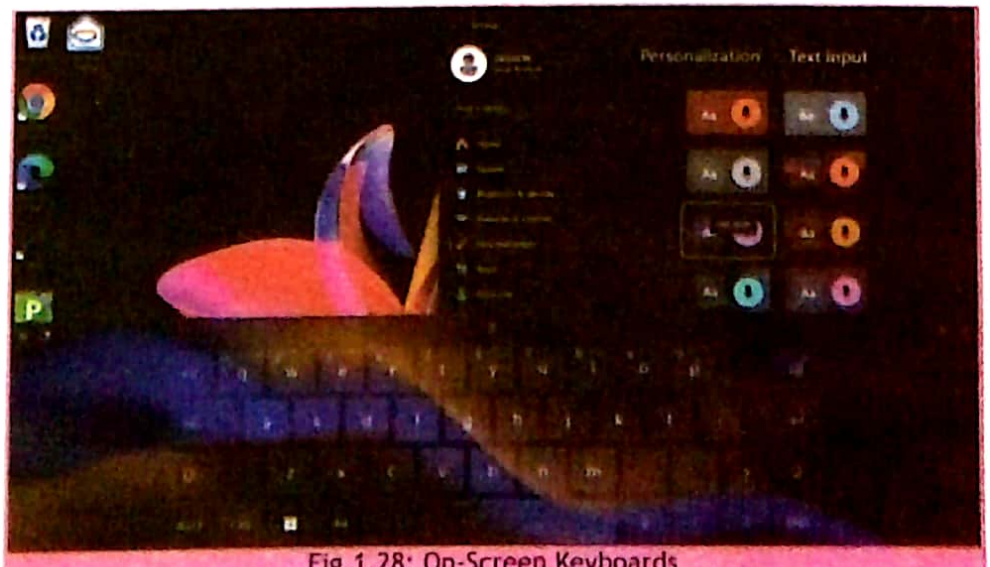


Fig.1.28: On-Screen Keyboards

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

1.2 Human-Computer Interaction (HCI)

Human-Computer Interaction (HCI) is a multidisciplinary field that studies how people interact with computers, digital devices, and software applications. The goal of HCI is to improve the interaction between users and systems by making these systems more user-friendly, efficient, and accessible. Good HCI can reduce user frustration, increase productivity, and ensure that technology meets the needs of its users.

Example: The design of a smartphone's touchscreen interface involves HCI principles to ensure that icons are large enough to be tapped easily, and that gestures (like swiping or pinching) are in-built.

1.2.1 Elements of Human-Computer Interaction

Human-Computer Interaction (HCI) is a complex and dynamic field that focuses on the design, evaluation, and implementation of interactive systems for human use. The effectiveness of any interactive system depends on several key elements that must be carefully considered during the design process. These elements include:

- Users
- Tasks
- Interface
- Environment

Users:

Users are the individuals who interact with a system. This involves studying their needs, abilities, limitations, and preferences. Users can vary widely in their skills, knowledge, experience, and expectations. Designers must consider the diversity of users, including factors such as age, cultural background, physical abilities, and technological proficiency.

Example: Designing a word processor for a professional writer involves understanding their need for robust editing tools, an organized workspace, and features that enhance their writing process, such as grammar checking, advanced formatting options, and distraction-free modes.

Tasks:

Tasks refer to the specific activities that users need to perform using the system. Identifying and understanding these tasks is essential for designing an interface that supports users in achieving their goals efficiently and effectively.

Example: On an e-commerce website, key tasks might include searching for products, filtering results, reading reviews, adding items to a shopping cart, and completing a purchase. Designers must ensure that each of these tasks is straightforward and easy to perform. As shown in Fig. 1.29.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

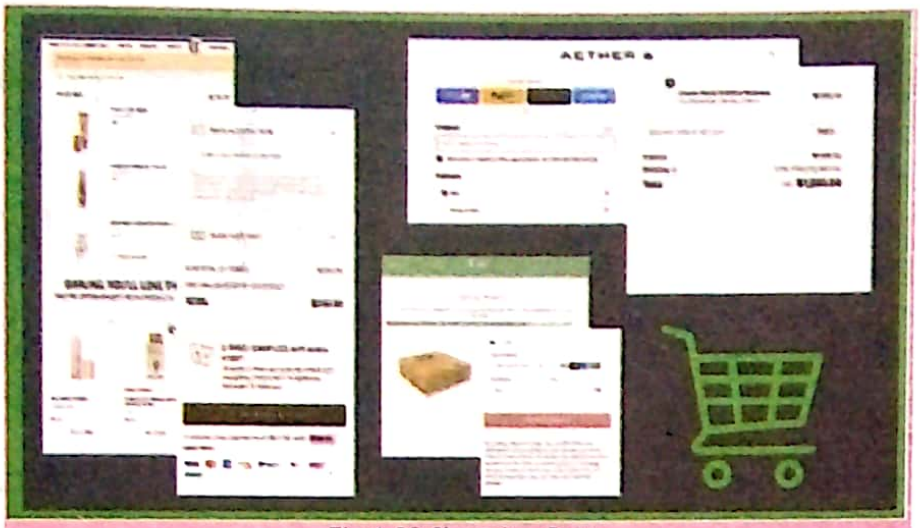


Fig.1.29 Shopping Cart

Interface:

The Interface is the point of interaction between the user and the system. It is where the design of the system comes to life, allowing users to interact with it through various input and output mechanisms. The interface can take many forms, including graphical user interfaces (GUIs), voice command systems, gesture-based interfaces, and more.

Example: The layout of a video editing software's interface is crucial for user productivity. Tools like cut, paste, and trim should be easily accessible, and the timeline should be carefully designed to allow for precise edits. A well-designed interface might also include customizable workspaces, where users can arrange tools and panels according to their workflow preferences. As shown in Fig.1.30.

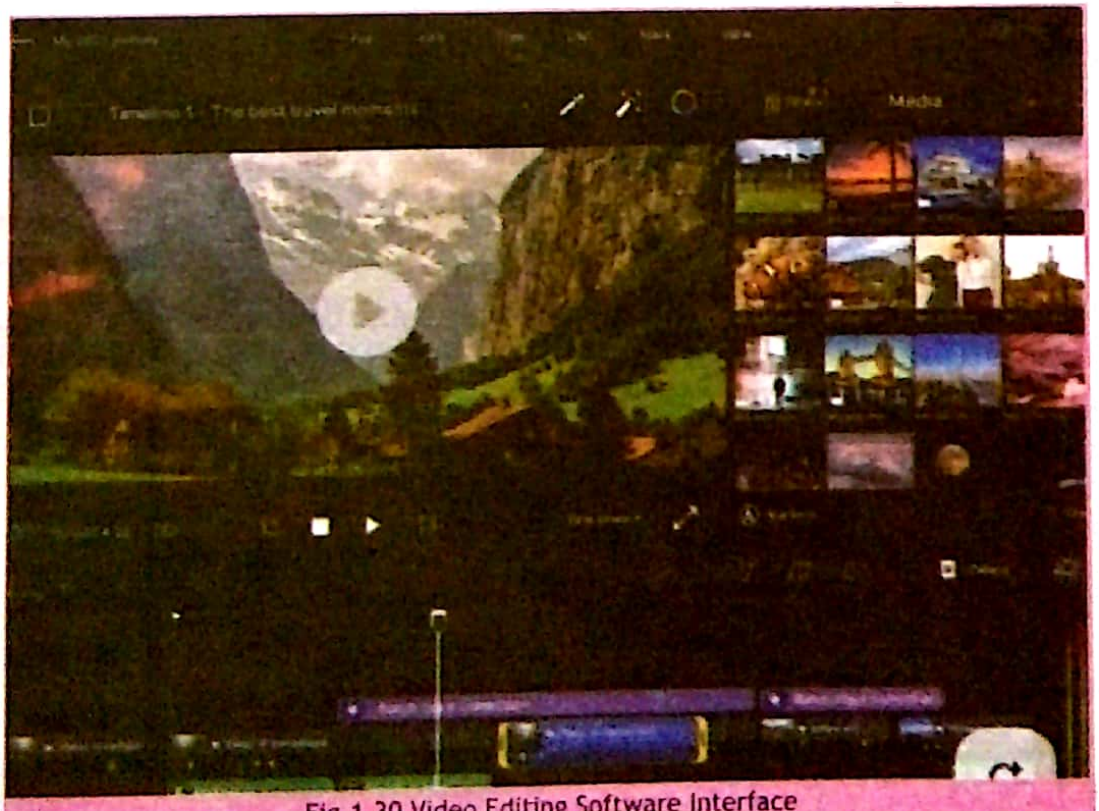


Fig.1.30 Video Editing Software Interface

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Environment:

The Environment refers to the physical and social context in which the interaction between the user and the system takes place. The environment can significantly influence how a system is used, and designers must consider these factors when creating interfaces. The environment includes the physical surroundings (e.g., lighting, noise levels, and available space) and the social context (e.g., whether the system will be used individually or collaboratively).

Fig.1.31: A Public Kiosk Interface

Example: A public kiosk interface, such as those found in airports or shopping malls, must be designed for quick, efficient use in potentially noisy, crowded environments. The interface should be simple, with large buttons and clear instructions to accommodate users who may be in a hurry or distracted by their surroundings. Additionally, the kiosk might need to be designed for use by standing users, meaning the screen should be at a comfortable height and angle. As shown in Fig. 1.31.

1.2.2 Usability in Human-Computer Interaction

Usability is a fundamental concept in Human-Computer Interaction (HCI), and it plays a crucial role in determining how effective and enjoyable a system is for its users. Usability refers to the extent to which a system can be used by specific users to achieve specified goals with effectiveness. A system with high usability is one that is easy to use and learn.

Key Components of usability in HCI:

Usability can be broken down into several key components, each of which contributes to the overall user experience in HCI. These components include Learnability, Efficiency, Memorability, Error Handling, and Satisfaction.

Learnability:

Learnability refers to how easy it is for new users to accomplish basic tasks the first time they encounter the system. A highly learnable system allows users to quickly grasp how to use it without needing extensive guidance or prior experience. This is especially important for applications or devices intended for a broad audience, where users may have varying levels of technical expertise.

Fig.1.32: Online Mobile Banking App

Example: Consider a mobile banking app designed for a diverse user base. A new user should be able to quickly learn how to navigate the app to check their account balance, transfer funds, or view transaction history. The use of onboarding tutorials or interactive guides can further enhance learnability, ensuring that users feel confident and capable from their first interaction as shown in Fig.1.32.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Efficiency:

Efficiency pertains to how quickly and effectively users can perform tasks once they have learned how to interact with the system. An efficient system allows users to achieve their goals with minimal time and effort.

Example: A professional graphic designer using design software like Adobe Photoshop needs to perform tasks such as editing images, creating layouts, and applying filters swiftly. Efficiency in this context means providing shortcut keys, customizable toolbars, and responsive tools that allow the designer to work with speed and precision.

Memorability:

Memorability refers to how easily users can remember how to use a system after a period of not using it. A system with high memorability ensures that users can return to it after some time and still remember how to perform key tasks without needing to relearn the interface.

Example: Imagine a user who only occasionally uses an online form submission system, such as a tax filing website. After several months, when they return to the site, they should be able to easily recall how to navigate the system, fill out the necessary forms, and submit their information. This can be facilitated through consistent design patterns, clear labeling, and a logical flow of steps.

Error Handling:

Error Handling involves the system's ability to prevent errors, guide users in correcting them, and help users recover from mistakes. Good error handling is essential for maintaining user trust and ensuring a smooth interaction, even when things go wrong.

Fig.1.33: Online booking system

Example: An online booking system (Fig.1.33) for airline tickets should prompt users to correct mistakes, such as entering an invalid date or leaving a required field blank. Instead of allowing the user to proceed with faulty data, the system might highlight the problematic fields in red and provide an error message explaining what needs to be corrected. Additionally, providing helpful suggestions or autofill options can prevent common errors, such as typos in names or addresses.

Useful Links!

Explore HCI Concepts: " Interaction Design Foundation offers free courses on HCI and usability principles."

Accessibility Tools: "Check out WebAIM for resources on making web content accessible to people with disabilities."

1.2.3 Common Problems in Human-Computer Interaction

Human-Computer Interaction (HCI) is a critical field that focuses on improving the interaction between users and computer systems. While significant progress has been made in designing

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

more user-friendly and efficient systems, several common problems still persist. These issues can negatively impact user experience, leading to frustration, errors, and even exclusion of certain user groups. Addressing these challenges is essential for creating systems that are accessible, user-friendly, and efficient for all users.

Interface Issues:

Poorly designed interfaces are one of the most common problems in HCI. The interface is the primary means through which users interact with a system, and its design can make or break the user experience. Common issues with interfaces include untidy layouts that could lead users to unclear navigation paths that make it difficult to locate important features. These problems can lead to confusion, inefficiency, and increased errors.

Example: Consider an e-commerce website with a poorly designed navigation menu. If the menu is untidy with too many options, users may struggle to find the categories they are looking for. For instance, a user searching for electronics might have to examine through an extensive list of unrelated categories, leading to frustration.

Accessibility Challenges:

Accessibility in HCI refers to the design of systems that can be used by people with a wide range of abilities and disabilities. However, many systems are not designed with accessibility in mind, leading to challenges for users with visual, auditory, motor, or cognitive impairments. These challenges often arise because designers do not fully consider the needs of all potential users.

Example: A website that lacks alternative text (Alt text feature) for images is inaccessible to users with visual impairments who rely on screen readers. These users might miss out on important content or context provided by images, making their experience incomplete or frustrating. Similarly, a mobile app that relies solely on sound for notifications may be unusable by individuals with hearing impairments.

Performance Issues:

Performance issues, such as slow or unresponsive systems, can severely impact the usability of a system. Users expect systems to respond quickly to their inputs, and any delay can lead to frustration and a decrease in productivity. Performance problems can be caused by a variety of factors, including poor system optimization, excessive resource use, or network latency. When systems are slow or fail to load content efficiently, users may abandon tasks or look for alternative solutions.

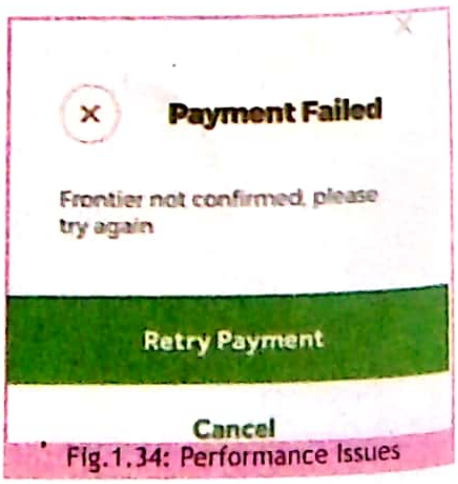


Fig.1.34: Performance Issues

Example: Imagine using an online banking app that takes an excessive amount of time to load account information or process transactions (Fig.1.34). Users may become impatient and concerned about the security of their transactions, leading them to avoid using the app in the future.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Poor Error Handling and Feedback:

Effective error handling and feedback are crucial components of a user-friendly system. When users encounter errors, they need clear, constructive feedback to understand what went wrong and how to fix it. Systems that provide poor or inadequate error messages can leave users confused and frustrated, leading to mistakes and inefficiency. Conversely, well-designed error handling helps users recover from mistakes quickly and with confidence, improving the overall user experience.

Example: Consider a form submission process on a website where the user is required to enter several fields of information. If the user leaves a required field blank or enters data in the wrong format, the system should provide an immediate, clear error message that explains what needs to be corrected. For instance, if an email field is left empty or filled incorrectly, the system should display a message like, **"Please enter a valid email address."** Without this guidance, users might not understand why their submission is not being accepted, leading to frustration and potentially abandoning the form altogether.

1.2.4. Methods for Improving Human-Computer Interaction

Improving Human-Computer Interaction (HCI) involves various strategies and methodologies aimed at creating systems that are more user-friendly, efficient, and complete. By addressing key aspects such as **user-centered design, accessibility, interface simplification, performance optimization, and error prevention,** developers can significantly enhance the users' experience.

User-Centered Design (UCD):

User-Centered Design (UCD) is a design philosophy that places the end users at the core of the design process. It is an iterative approach where user feedback is continuously gathered and integrated throughout the development cycle. This method ensures that the final product aligns closely with the users' needs, preferences, and limitations, resulting in a more effective and satisfying user experience.

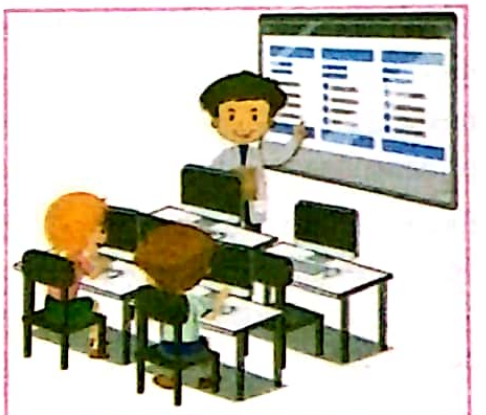


Fig.1.35: User-Centered Design (UCD)

Example: Consider the development of a new educational software intended for use in schools. By involving teachers and students in the design process from the beginning, developers can gather valuable feedback on features such as interactive elements, user interfaces, and content organization. Early testing might reveal that students prefer interactive quizzes over static reading material, leading to adjustments that enhance engagement and learning outcomes. As shown in Fig.1.35.

Accessibility Enhancements:

Accessibility enhancements focus on making systems usable for individuals with a range of disabilities. This involves designing systems that are perceivable, practicable, understandable, and user-friendly, ensuring that people with visual, auditory, motor, or cognitive impairments can effectively interact with the system.

Example: An educational platform that includes captions and transcripts for all video content

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

improves accessibility for students with hearing impairments. Additionally, incorporating keyboard navigation and screen reader compatibility ensures that users with motor impairments or visual disabilities can interact with the platform effectively.

Simplifying User Interfaces:

Simplifying user interfaces involves reducing unnecessary complexity and presenting information in a clear, concise manner. A well-designed interface should prioritize essential features and streamline interactions to minimize user effort and cognitive load.

Example: A streamlined email application that features a minimalist design with only the most commonly used tools (such as compose, inbox, and search) can significantly enhance user efficiency. By avoiding complex menus and excessive options, users can quickly access and manage their emails without unnecessary distractions.

Improving Performance

Optimizing system performance is essential for ensuring that applications run smoothly and respond promptly to user inputs. Performance issues such as slow load times or unresponsive interfaces can detract from the user experience and hinder productivity.

Example: Reducing the load time of a large e-commerce website by optimizing images, compressing files, and implementing efficient coding practices can greatly improve user experience.

Error Prevention and Recovery:

Effective error prevention and recovery mechanisms are crucial for minimizing the impact of user mistakes and system errors. Systems should be designed to prevent errors whenever possible and provide clear feedback and recovery options when errors do occur.

Example: A spreadsheet application that prompts users to save changes before closing the file helps prevent data loss. Additionally, an online form with real-time validation checks for required fields and provides immediate feedback when errors are detected can help users correct mistakes before submission.

1.2.5 Ethical, Social, Economic, and Environmental Implications in HCI

Human-Computer Interaction (HCI) is not just about creating systems that are user-friendly and efficient; it also encompasses broader considerations that affect society at large. Ethical, social, economic, and environmental implications play a crucial role in the development and deployment of any new system. Understanding these aspects is essential for designing systems that are not only effective but also responsible and sustainable.

Ethical Implications:

Ethical implications in HCI address how technology affects users' rights, privacy, and dignity. Ethical design ensures that technology is developed and used in ways that are respectful and fair, protecting users from harm and misuse.

- **Data Privacy:** One of the major ethical concerns is how user data is collected, stored, and

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

used. Developers must ensure that user data is handled with confidentiality and used only for its intended purpose. Users should have control over their data and be informed about how it is used.

Example: Social media platforms such as, Instagram, Facebook, WhatsApp and Twitter (Fig.1.36), collect vast amounts of personal data from users. Ethical practices require these platforms to implement strong data protection measures, obtain explicit user consent before collecting data, and provide clear options for users to manage their privacy settings.

Fig.1.36: Social media platforms

- **User Consent:** Users should be informed and provide consent before engaging with systems that collect or use their data. Consent should be clear and informed.

Example: Mobile apps that request access to personal information, such as location or contacts, should provide users with a clear explanation of why this access is necessary and obtain explicit consent before proceeding.

Social Implications:

Social implications refer to how technology influences communication, social interactions, and access to information. HCI can have both positive and negative effects on social structures and relationships.

For Your Info!

Social Media and Mental Health: "Research has shown that excessive use of social media can lead to anxiety and depression, particularly among young people. Balanced use is a key."

Example: Smartphones and social media enable instant messaging and video calls, fostering global communication. However, excessive screen time and reliance on digital interactions can affect face-to-face communication skills and lead to issues such as social isolation.

Economic Implications:

Technology impacts the economy by creating new opportunities and challenges. Understanding these implications helps in designing systems that support economic growth while addressing potential disruptions.

Example1: The rise of e-commerce platforms like Amazon (Fig.1.37) has generated jobs in tech, logistics, and customer service. These platforms have also led to the growth of related industries, such as digital marketing and online payment systems.

Fig.1.37: Amazon

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Example2: The automation of manufacturing processes through robotics (Fig.1.38) can lead to increased production efficiency but may also result in job losses for assembly line workers. Training and reskilling programs can help mitigate these effects.

Fig.1.38: Manufacturing through robotic

Environmental Implications:

The environmental impact of technology involves considering energy consumption, resource use, and waste management. Sustainable practices in HCI are crucial for minimizing the ecological footprint of technology.

Example1: Companies like Google and Microsoft are investing in renewable energy to power their data centers, aiming to reduce their carbon footprint and promote sustainability.

Example2: Electronics manufacturers are implementing take-back programs and designing products with recyclable materials to reduce e-waste. Initiatives such as Apple's recycling program help ensure that old devices are properly recycled.

1.3 Trade-offs between Usability and Security in Computing Systems

When designing computing systems, achieving a balance between usability and security is crucial. Both factors are essential, but they often come into conflict. The following is a detailed look at the trade-offs between usability and security, along with recommendations for cybersecurity measures considering efficiency, cost, privacy, and ethics.

1.3.1 Trade-offs between Usability and Security

Usability focuses on how easy and intuitive a system is for users, while **security** aims to protect the system from unauthorized access and attacks. The trade-offs arise because enhancing security can sometimes degrade usability, and vice versa.

The following table represents the trade-offs between usability and security in computing systems, along with examples.

Aspect	Security	Usability	Example
Authentication Complexity	Enhances security by adding extra verification steps like multi-factor authentication (MFA), making unauthorized access much harder.	Reduces usability as MFA requires additional steps like entering codes from a phone, slowing down user access and causing potential frustration.	A user logging into a bank account needs to enter a password and then a code sent to their phone, increasing security but delaying access.
Access Controls	Protects sensitive data by restricting user permissions, ensuring only authorized individuals can access or modify information.	Hinders usability when users frequently encounter permission issues, requiring them to request access, which can slow down their work.	An employee needs frequent permission from IT to access certain files, delaying their ability to complete tasks efficiently.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Aspect	Security	Usability	Example
Encryption	Secures data by converting it into a format only accessible with a decryption key, protecting against unauthorized access even if data is intercepted.	Impacts usability as encryption can slow down data access and processing, and managing encryption keys can be complex and time-consuming.	Encrypting patient records keeps them secure but may slow down access, which can be challenging during emergencies.
Regular Updates and Patches	Maintains security by fixing vulnerabilities and protecting against new threats through regular updates.	Disrupts usability as updates may require system restarts or interrupt work, and they can sometimes introduce compatibility issues.	Frequent software updates may interrupt employees' work, requiring them to restart their systems, which can be inconvenient during busy times.

1.3.2 Recommendations for Cybersecurity measures

When designing and implementing cybersecurity measures, it is important to consider various factors to strike a balance between security and user experience. The key factors to consider are efficiency, cost, privacy, and ethics. Each of these plays a crucial role in determining how effective and sustainable the security solutions will be.

Efficiency: Efficiency in cybersecurity involves implementing solutions that protect systems without significantly hindering user productivity. The following computing solutions illustrate the principle of efficiency in cybersecurity measures.

- **Auto-Save and Backup Systems:**

Automatically saving work and backing up files regularly helps ensure that data is not lost in case of a system crash or cyberattack. This increases efficiency by reducing the risk of lost work and the need to redo tasks.

Example: Google Docs automatically saves documents in real-time to Google Drive, so users don't have to worry about losing their work if their computer shuts down unexpectedly.

- **Password Managers:**

Password manager stores and autofill strong, unique passwords for different websites, reducing the time and effort users spend managing multiple passwords while enhancing security.

Example: Using a password manager like LastPass or Bitwarden, users can automatically log into websites without needing to remember complex passwords, making the login process quicker and more secure.

Cost: Cost considerations are critical when determining the scope and scale of cybersecurity measures. The goal is to implement effective security within budget constraints. The following computing solutions illustrate the principle of cost in cybersecurity measures.

Helping Tip!

Choosing Security Tools: Consider open-source security tools as a cost-effective alternative to proprietary software, especially for small businesses or personal use.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Freemium Security Software:

Many security software providers offer free basic versions of their products, which can be sufficient for small businesses or individual users, allowing them to secure their systems without incurring high costs.

Example: Avast offers a free version of its antivirus software that provides essential protection against viruses and malware, which is often adequate for home users.

Cloud-Based Security Services:

Cloud-based security solutions often have lower upfront costs compared to on-premises hardware, as they eliminate the need for expensive infrastructure and maintenance.

Example: A small business might choose to use a cloud-based firewall service, such as Cloudflare, to protect their website from attacks, avoiding the cost of purchasing and maintaining their own hardware.

Open Source Solutions:

Open source tools and frameworks offer cost-effective alternatives to expensive proprietary software, providing robust security features without significant financial investment.

Example: The OWASP (Open Web Application Security Project) provides a variety of open-source tools that help developers identify and mitigate security vulnerabilities in web applications, reducing the need for expensive commercial solutions.

Privacy: Privacy protection is a foundation of cybersecurity, ensuring that users' personal data is handled responsibly and securely. The following computing solutions illustrate the principle of privacy in cybersecurity measures.

Private Browsing Modes:

Private browsing, or "incognito" mode, in web browsers prevents the storage of browsing history, cookies, and other data, helping protect user privacy on shared or public computers.

Example: When using a public computer at a library, users can enable incognito mode in Chrome or Firefox to ensure their browsing activity isn't stored, protecting their privacy.

Anonymization and Encryption:

These techniques protect user privacy by ensuring that even if data is accessed by unauthorized parties, it cannot be traced back to individuals or easily deciphered.

Example: Healthcare providers often encrypt patient records and anonymize data before sharing it for research purposes, ensuring that personal health information remains confidential.

Two-Factor Authentication (2FA):

2FA adds an extra layer of security by requiring users to provide two forms of identification (such as a password and a code sent to their phone), which helps protect their accounts from unauthorized access. As shown in Fig. 1.39.

Fig.1.39: Two-Factor Authentication (2FA)

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Example: When logging into their email, a user might enter their password and then receive a text message with a one-time code, ensuring that only they can access their account even if someone else knows their password.

Ethics: Ethical considerations in cybersecurity involve respecting user rights and maintaining trust through transparent and fair practices. The following computing solutions illustrate the principle of ethics in cybersecurity measures.

User-Friendly Privacy Settings:

Designing privacy settings that are easy to understand and adjust ensures that all users, regardless of their technical knowledge, can control how their data is used.

Example: Facebook provides a simple privacy checkup tool that guides users through their privacy settings step by step, making it easy to adjust who can see their posts and personal information.

Accessibility Features for Disabled Users:

Ensuring that security measures are accessible to users with disabilities is an ethical responsibility, helping to prevent exclusion from digital services.

Example: An online banking platform might offer voice recognition for account access, making it easier for users with physical disabilities to log in securely without needing to type a password as shown in Fig.1.40.

Fig.1.40: Online Banking-Voice Recognition

Useful Links

Cybersecurity and Usability Research: Usable Security and Privacy Group at Carnegie Mellon University explores ways to design systems that are both secure and user-friendly.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Usability:** Measures how effortlessly and effectively users can interact with a system.
- **Moderated Usability Testing:** Involves a facilitator guiding and observing users.
- **Unmoderated Usability Testing:** Users complete tasks independently, often remotely.
- **Qualitative Usability Testing:** Focuses on user experiences and feedback.
- **Quantitative Usability Testing:** Measures usability with numerical data.
- **Remote Usability Testing:** Testing conducted from the user's location.
- **A/B Testing:** Compares two versions of a system to evaluate performance.
- **Complex Interfaces:** Overwhelming users with too much information or complexity.
- **Inconsistent Navigation:** Variability in navigation elements across the system.
- **Poor Error Messages:** Unclear or unhelpful error messages.
- **Lack of Accessibility:** Failure to accommodate users with disabilities.
- **Security:** Measures to protect information and systems from unauthorized access and threats.
- **Malware:** Malicious software designed to harm systems.
- **Phishing:** Deceptive attempts to obtain sensitive information.
- **Man-in-the-Middle (MitM) Attacks:** Interception and potential alteration of communication.
- **Denial of Service (DoS) Attacks:** Overloading a system to make it unavailable.
- **Insider Threats:** Security risks from within an organization.
- **Accessibility:** Design to ensure usability by people with a range of abilities and disabilities.
- **Visual Impairments:** Solutions like screen readers, high contrast modes, and zoom functions.
- **Hearing Impairments:** Solutions like closed captions, transcripts, and sign language interpretation.
- **Motor Impairments:** Solutions like voice recognition software, alternative input devices, and on-screen keyboards.
- **Human-Computer Interaction (HCI):** A multidisciplinary field studying interactions between people and computers to make systems more user-friendly, efficient, and accessible.
- **Users:** Individuals interacting with a system; their needs, abilities, limitations, and preferences must be considered.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- **Tasks:** Specific activities users perform using the system.
- **Interface:** The point of interaction between the user and the system, including GUIs, voice commands, and gesture-based controls.
- **Environment:** The physical and social context of the interaction, such as lighting, noise, and whether the use is solitary or collaborative.
- **Interface Issues:** Problems with poorly designed interfaces, such as untidy layouts and unclear navigation.
- **Accessibility Challenges:** Issues in making systems usable for people with disabilities, including visual, auditory, motor, or cognitive impairments.
- **Performance Issues:** Problems such as slow or unresponsive systems affecting usability.
- **Poor Error Handling and Feedback:** Inadequate error messages and feedback leading to user frustration.
- **User-Centered Design (UCD):** A design approach focusing on users' needs and feedback throughout the development cycle.
- **Accessibility Enhancements:** Designing systems to be usable by people with various disabilities.
- **Simplifying User Interfaces:** Reducing complexity and presenting information clearly.
- **Improving Performance:** Optimizing system performance to ensure smooth operation.
- **Error Prevention and Recovery:** Mechanisms to prevent and recover from errors effectively.
- **Ethical Implications:** Concerns about user rights, privacy, and dignity.
- **Social Implications:** The impact of technology on communication and social interactions.
- **Economic Implications:** The effect of technology on the economy, including job creation and disruption.
- **Environmental Implications:** Considering energy consumption, resource use, and waste management in technology.
- **Authentication Complexity:** Balancing security measures like multi-factor authentication with user convenience.
- **Access Controls:** Restricting data access to protect security while considering the impact on user workflow.
- **Encryption:** Protecting data with encryption while managing the potential impact on performance.
- **Regular Updates and Patches:** Maintaining system security through updates while managing disruptions to usability.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. A software product receives negative user reviews due to being difficult to use. Which of the following factors is least likely responsible?
   a. Inefficiency b. Ineffectiveness c. High satisfaction d. High complexity
2. A web form alerts the user when an email address is entered incorrectly before submission. Which usability principle does this best represent?
   a. User satisfaction b. Error prevention
   c. Performance optimization d. Feedback mechanism
3. If you want to observe users' emotional reactions and gather verbal feedback while they perform tasks, which usability testing method would be most suitable?
   a. Moderated Usability Testing b. A/B Testing
   c. Unmoderated Usability Testing d. Remote Usability Testing
4. Which practice best supports confidentiality in a hospital's patient records system?
   a. Ensuring the system is always online
   b. Allowing only authorized doctors to access patient data
   c. Letting patients edit their own files freely
   d. Encrypting data for public access
5. A user receives a seemingly legitimate email from their bank requesting login details. This is an example of:
   a. Phishing b. Malware c. Spyware d. Spoofing
6. A company wants to protect client data during transmission over the internet. Which security method should they apply?
   a) Firewall b. Encryption c. Access Control d. Antivirus Software
7. What is the main objective of Human-Computer Interaction (HCI)?
   a. System performance b. User interaction
   c. Software complexity d. Hardware costs
8. Which of the following is NOT considered a key element of HCI?
   a. Users b. Tasks c. Budget d. Interface
9. Which of the following is a common issue related to accessibility in HCI?
   a. Slow load times b. Inadequate error messages
   c. Lack of alternative text for images d. Complicated authentication processes
10. Which method involves placing the end users at the core of the design process in HCI?
    a. Performance Optimization b. Error Prevention
    c. User-Centered Design (UCD) d. Simplifying User Interfaces
11. What does the term 'interface' refer to in HCI?
    a. The physical device used b. Point where user interacts with system
    c. The software programming language d. The hardware configuration

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

12. Why is multi-factor authentication (MFA) often seen as a trade-off between security and usability?
    a. It's expensive to implement
    b. It improves system speed
    c. It increases security but can slow down user access
    d. It reduces the need for encryption
13. Which issue is caused by encryption?
    a. Processing speed b. Data access c. Key management d. Interface
14. How can auto-save improve cybersecurity efficiency?
    a. Recovery b. Encryption c. Complexity d. Management
15. Which ethical consideration involves transparency in data use?
    a. Efficiency b. Cost c. Privacy d. Accessibility

Give short answers to the following Short Response Questions (SRQs).

1. What are the key components of usability in HCI?
2. List and briefly describe the four key elements of HCI.
3. What is the difference between moderated and unmoderated usability testing?
4. What does 'error handling' mean in the context of HCI?
5. What is meant by 'User-Centered Design (UCD)' in HCI?
6. Give three benefits and two drawbacks of using password managers?
7. How does user feedback contribute to effective system design?
8. Provide an example of how user feedback led to a design change.
9. Give two considerations for designing accessible systems for users with disabilities.
10. Suggest three features that should be included in a new educational software application to enhance accessibility for users with disabilities.

Give long answers to the following Extended Response Questions (ERQs).

1. Describe how accessibility challenges impact the usability of a digital system and provide examples of improvements that can address these challenges.
2. What are the main elements involved in Human-Computer Interaction (HCI) design? Explain each element's role in creating effective interactive systems.
3. Explain the ethical considerations in Human-Computer Interaction (HCI) with examples. Why are these considerations important in the design and deployment of computing systems?
4. You are developing a new software tool for remote team collaboration. What usability and security features would you integrate to support efficient teamwork while protecting sensitive data?
5. You are tasked with designing an e-commerce website. The site must be both user-friendly and secure. List three features you would include to achieve this balance and explain why.
6. Discuss the trade-offs between usability and security in computing systems. Provide examples of how these trade-offs might manifest in real-world applications.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

7. You are tasked with designing an e-commerce website. The site must be both user-friendly and secure. List three features you would include to achieve this balance and explain why.
8. How do, security measures like multi-factor authentication (MFA) and encryption impact usability, and how can designers balance these trade-offs?

Activity 1: Usability and Accessibility Design Challenge

Objective: To understand and apply principles of usability and accessibility in system design.

Group Project: Divide students into small groups and task them with designing a mock interface for a hypothetical app that must be user-friendly and accessible to people with disabilities.

Instructions:

- Choose an application (e.g., a fitness tracker, e-learning platform, or social media app).
- Apply usability principles (simplicity, consistency, feedback, error prevention, learnability) to the design.
- Incorporate accessibility features (e.g., voice guidance, large buttons, high-contrast modes).
- Create wireframes or mockups of the interface.
- Present the design to the class and explain how the features support both usability and accessibility.

Activity 2: Security Measures Evaluation

Objective: To analyze the impact of different security measures on system usability and functionality.

Individual Research and Presentation: Students research and present on different security measures and their potential impacts on usability.

Instructions:

- Choose one security measure (e.g., multi-factor authentication, encryption, or firewalls).
- Explain how this measure works and its purpose.
- Discuss how it might affect the usability of a system (e.g., increased steps for user login, potential for user confusion).
- Provide real-world examples of systems where this trade-off is evident.
- Present findings to the class and suggest ways to mitigate usability issues while maintaining strong security.

Activity 3: Design an Inclusive Interface

Objective: Students will design a user interface for a specific target audience considering different types of disabilities.

Instructions:

- **Identify User Needs:** Divide students into groups and assign each group a different disability (visual, auditory, motor).
- **Design Brief:** Each group will design an interface for a common application (e.g., a public information kiosk, a health tracking app) that caters to their assigned disability.
- **Design Elements:** Ensure designs include features like screen readers, high contrast modes, alternative input methods, etc.
- **Presentation:** Groups present their designs to the class, explaining how their interface meets accessibility standards.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Activity 4: Usability Testing Simulation

Objective: To practice and understand different usability testing methods.

Simulation Exercise: Conduct a simulated usability test of a given website or app, applying different testing methods.

Instructions:

1. **Preparation:** Select a website or app to test.
2. **Testing Methods:** Use different types of usability testing methods (moderated, unmoderated, qualitative, quantitative) to evaluate the system. For example:
   - **Moderated Test:** Have a classmate use the system while you observe and take notes.
   - **Unmoderated Test:** Have another student perform tasks on their own and record their interactions.
   - **Qualitative Test:** Conduct interviews with users to gather their feedback.
   - **Quantitative Test:** Measure task completion times and error rates.
3. **Analysis:** Compile the results from different tests and identify common usability issues.
4. **Reporting:** Prepare a report summarizing the findings and suggest improvements based on the testing.

Activity 5: Evaluating and Improving an Educational App's Interface

Suggested App: Taleemabad

- Taleemabad is a popular educational app aimed at teaching children in a fun and interactive way. It provides engaging content in subjects like mathematics, English, and Urdu through games and animated lessons.

Short Activity Overview:

Evaluate the Taleemabad app's interface, identify areas of improvement, and suggest user-centered changes that could enhance learning and usability.

Questions for Evaluation:

1. Is the app easy to navigate for both children and parents?
2. Is the app's design colorful and engaging for children?
3. Does the design support ease of use, or is it overwhelming?
4. Can children easily start and finish a lesson?
5. Is the app's flow intuitive for young users without adult assistance?

Suggestions for Improvement:

1. What would you suggest to make the app more user-friendly?
2. Propose 1-2 design changes that would improve the app's usability for young learners.

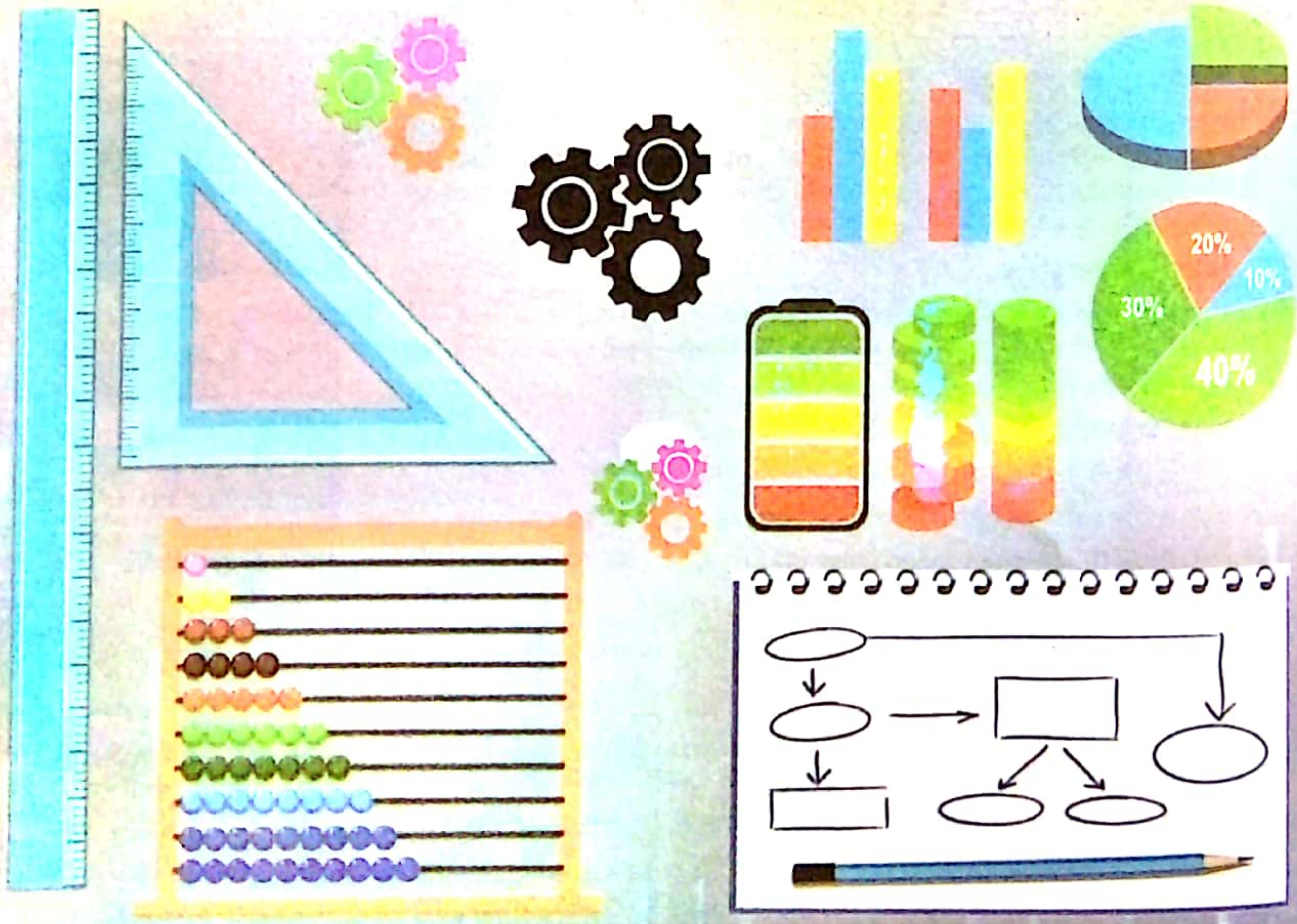
Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

2 Computational Thinking & Algorithms

Learning Outcomes

At the end of this unit students will be able to:

- Identify and apply complex algorithms on data structures such as trees and binary search
- understand and evaluate the computational solutions in terms of efficiency , clarity and correctness



Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Introduction

Computational Thinking (CT) is a problem-solving process rooted in computer science principles. It enables individuals to tackle complex challenges by breaking them down into manageable parts and approaching them logically and systematically. Although CT is fundamental to computer science, its application extends far beyond – into science, engineering, business, medicine, and everyday life.

The core pillars of computational thinking include:

- **Decomposition:** breaking a problem into smaller, more manageable components.
- **Pattern Recognition:** identifying similarities or trends to simplify complex tasks.
- **Abstraction:** focusing on important information and ignoring irrelevant details.
- **Algorithm Design:** developing clear, step-by-step instructions to solve a problem.

Together, these elements form the foundation for designing effective and efficient computational solutions. At this advanced level, Computational thinking is not only applied to design algorithms but also is used to analyze and evaluate these solutions in terms of correctness, clarity, and efficiency. Understanding data structures is central to this process, as they greatly influence how well a solution performs.

In this chapter, it will be explored how computational thinking integrates with algorithms and data structures to create solutions that are not only correct but also optimized and scalable.

2.1 Data Structures

Data structure defines the way for organizing and storing data so that it can efficiently be accessed and modified. They provide a framework for managing and organizing data in computer programs, allowing for various operations to be performed effectively.

Data structures are essential for efficiently organizing and managing data in computer programs. They dictate how data is arranged in memory and impact the performance of various operations like insertion, deletion and retrieval. For instance, arrays provide quick access to elements by index, while linked lists offer flexibility in dynamic data manipulation. The choice of data structure e.g. stacks and queues, trees and graphs depends on the specific needs of the application and the operations it performs.

Following are some common data structures:

2.1.1 Arrays

Arrays are the simplest data structure used in programming that contain different elements of same size. Contiguous memory locations are used to store these elements. The memory locations are identified by indexes as shown in Fig.2.1.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

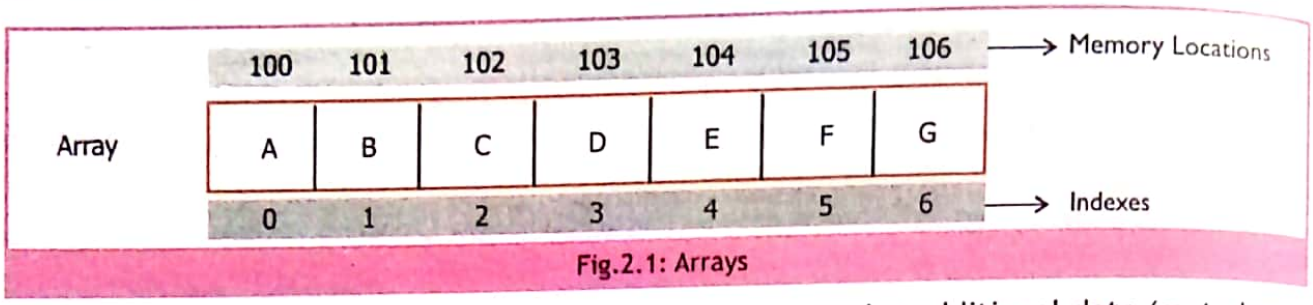


Fig.2.1: Arrays

The arrays are considered efficient because they do not require additional data (metadata) to store the elements. Additionally, because the memory locations are contiguous, therefore, it makes accessing the elements fast.

However, the arrays are fixed in size and it requires to declare its size at the start. At some later stage, when arrays need to be resized, it is costly to do so, because it may require creating new arrays with larger size and shifting of elements from old array to new. Similarly, the arrays could not be a good choice when elements are of different type.

The arrays are of two different types:

One-Dimentional (1-D) Array: In 1-D array , various elements (of same type) are stored in a single list. The element can be reached by knowing only a single index. One-Dimentional (1-D) Array example is shown in Fig.2.2.

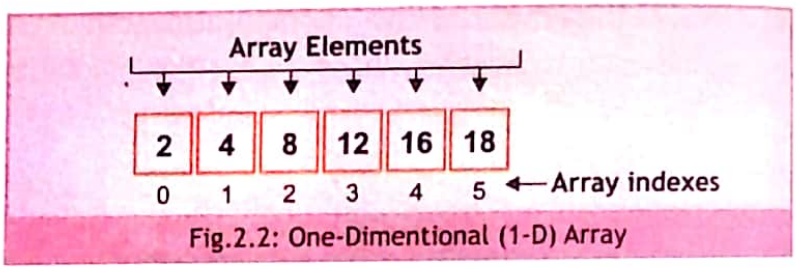


Fig.2.2: One-Dimentional (1-D) Array

Two-Dimentional (2-D) Array: The 2-D array is also known as matrix, where the elements are stored in the form of matrix or table. The elements can only be reached by knowing two indexes e.g. row and column. Fig.2.3 shows an example of Two-Dimentional (2-D) Arrays.

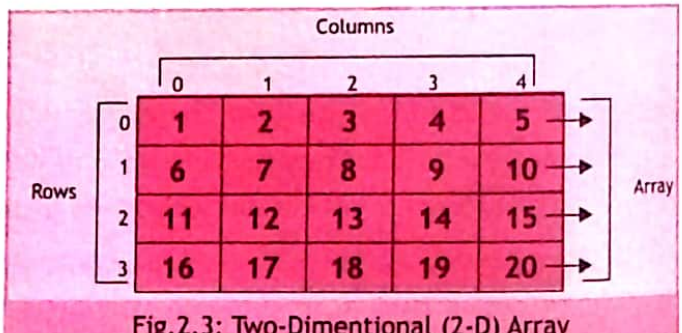


Fig.2.3: Two-Dimentional (2-D) Array

2.1.2 Linked Lists

Linked Lists are basic data structures in which elements (called nodes) are connected to each other using the concept of pointers. The nodes contain the values to be stored and pointer is the reference (memory address) of the next node in the sequence as shown in Fig.2.4.

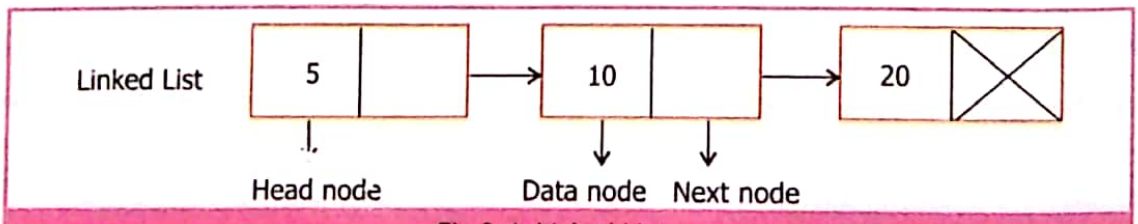


Fig.2.4: Linked Lists

The linked lists are capable to store different types of data and it also makes insertion and deletion operations easy as compared to arrays. The insertion/deletion could be performed at

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

any position in the sequence of linked list. Another advantage of linked lists is that they are dynamic in size (no need to specify the size of linked list at the start, rather they can grow and shrink as per requirement.

Singly Linked List

Singly linked list is basic and simple type of list, it maintains two parts in each node.

- data to be stored
- reference to next node in the sequence

Fig.2.5 shows an example scenario:

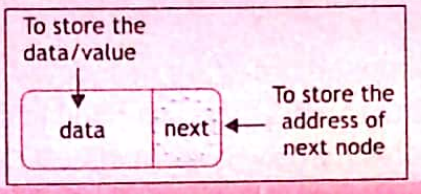


Fig.2.5: singly linked list node structure

The last node of the linked list contains NULL because it does not refer to any other node as shown in Fig.2.6.

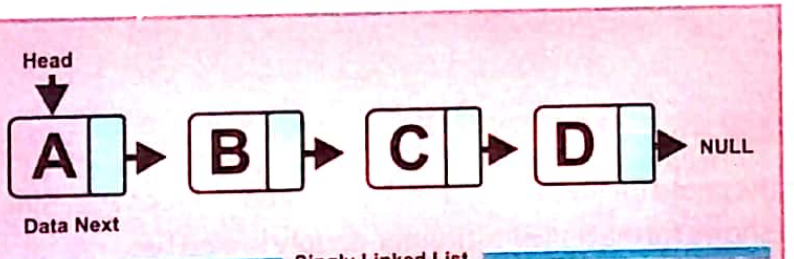


Fig.2.6: Singly Linked List

Doubly Linked List

The doubly linked list is complex as compared to singly linked list but it also provides more advantages. The major advantage is that it allows traversal in both directions and for that purpose it needs to maintain pointers for both directions e.g. reference to previous node and reference to next node. Therefore, the node in doubly linked list have three parts (Fig.2.7):

- reference to previous node in the sequence
- data to be stored
- reference to next node in the sequence

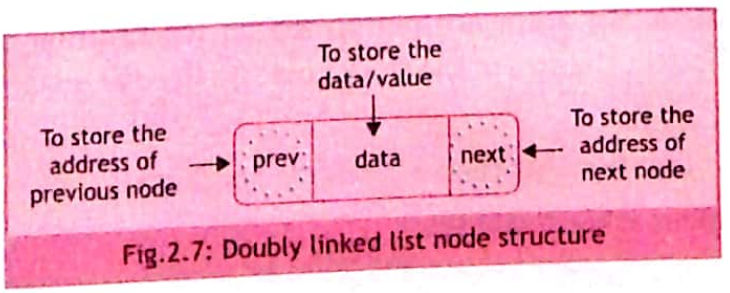


Fig.2.7: Doubly linked list node structure

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Fig.2.8 shows an example scenario:

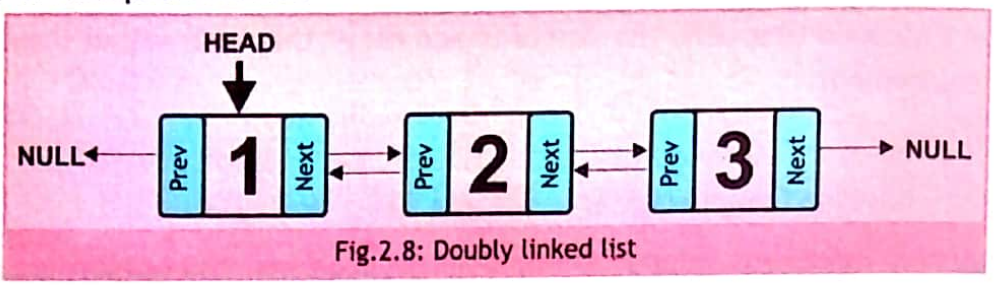


Fig.2.8: Doubly linked list

The first node and the last node in the sequence of linked list contains NULL because it does not refer to any other nodes.

Circular Linked List

It is a special type of linked list where all nodes are connected in a circle - forming a loop. Unlike other types, where last node points to NULL, the last node in circular linked linked is connected back to the first node. Therefore, while traversing in circular linked linked, it never reached to NULL. The structure of Circular Link List is depicted in Fig.2.9.

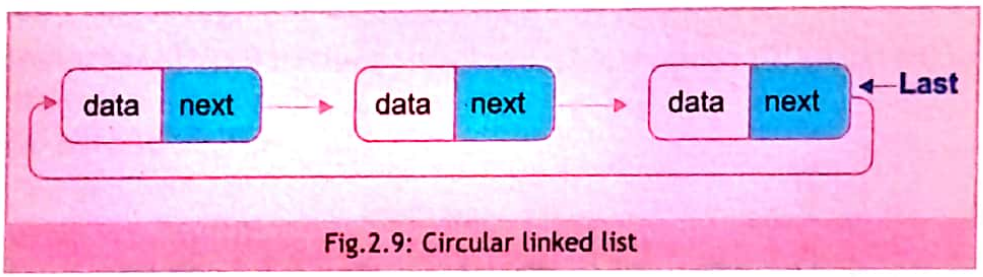


Fig.2.9: Circular linked list

The circular linked list could be made from both of the above types e.g. Singly linked list or doubly linked list. Fig.2.10 shows formation of a circular doubly linked list.

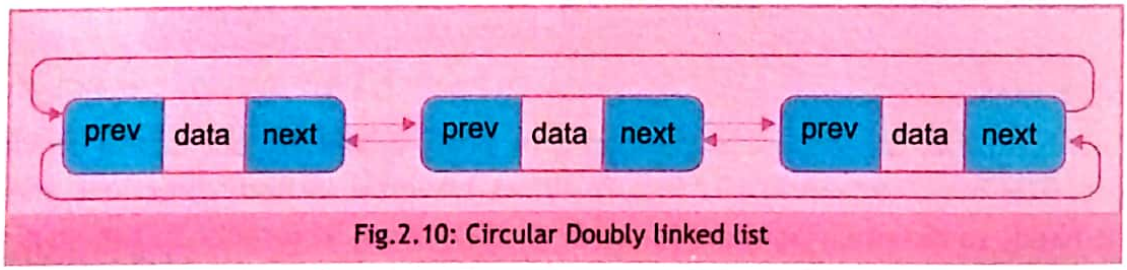


Fig.2.10: Circular Doubly linked list

2.1.3. Stacks

The stacks use the LIFO (Last-In, First-Out) principle, where the most recently added element (last element in the atack) is the first to be removed. This is important for understanding algorithms that need to manage tasks in a specific order. The LIFO principle makes it simple because the operations on the stack are performed on top position of the stack. Stack are considered memory efficient, the size of the stack could be fixed or dynamic, it depends upon the implementation.

Following are the basic operation that can be performed on stack:

- Push: Add a new element at the top of stack
- Pop: remove existing element from the top of stack

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- Top: return the top element without removing it
- IsEmpty: Check, if stack is empty
- IsFull: Check, if stack is full

The working of Stack with core operations is shown Fig.2.11.

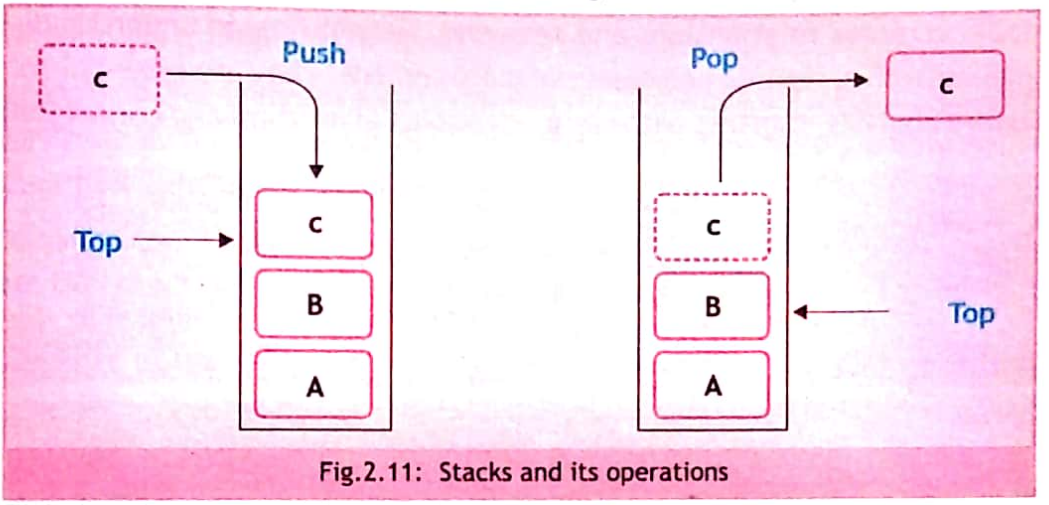


Fig.2.11: Stacks and its operations

2.1.4. Queues

The Queues use the FIFO (First-In, First-Out) principle, where the first element added is the first to be removed. This is useful for managing tasks that need to be processed in the order they arrive. Unlike stack, the operations on queues occur at two ends. The new elements are added at the end of queue, whereas the existing elements are removed from the start of the queue.

Following are the basic operation that can be performed on queues:

- Enqueue: Add a new element at the end(back/tail/rear) of the queue
- Dequeue: remove existing element at the start (front/head) of the queue.
- Front: return the start element without removing it
- IsEmpty: Check, if queue is empty
- IsFull: Check, if queue is full

The working of Queues with core operations is shown Fig.2.12.

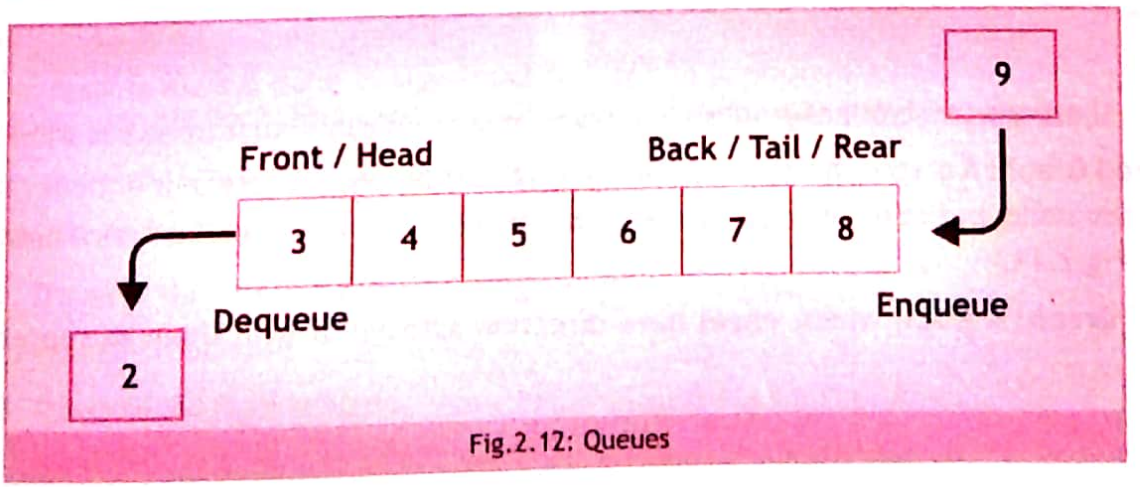


Fig.2.12: Queues

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

They are key consideres key data structure in algorithms like breadth-first search (BFS) and in various real-world scenarios like scheduling and buffering.

2.1.5. Graphs

The graph data structure also consist of nodes (called vertices) connected to each other by edges.

The graphs model complex relationships and networks. Graphs help in understanding problems related to social networks, routing and resource management. They are essential for algorithms related to network analysis, shortest paths (e.g., Dijkstra's algorithm) and connectivity as shown in Fig.2.13.

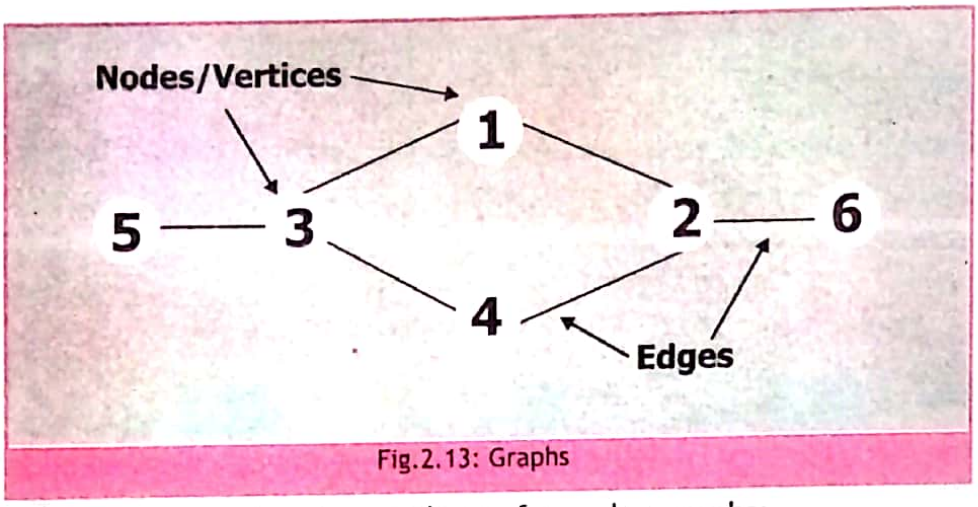


Fig.2.13: Graphs

Following are the basic operation that can be performed on graphs:

- Add Vertex: Add a new node to the graph
- Add Edge: Create connection between two nodes
- Remove Vertex/Edge: Delete node or edge from the graph
- Traversal: Visiting graph in specific order.

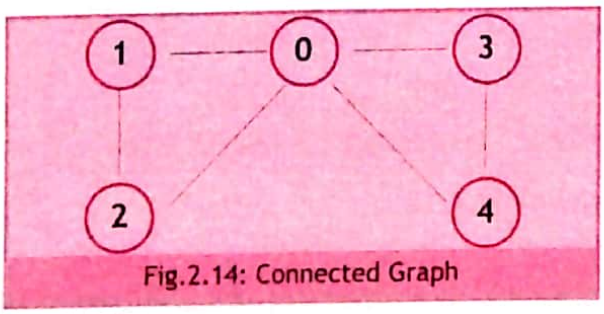


Fig.2.14: Connected Graph

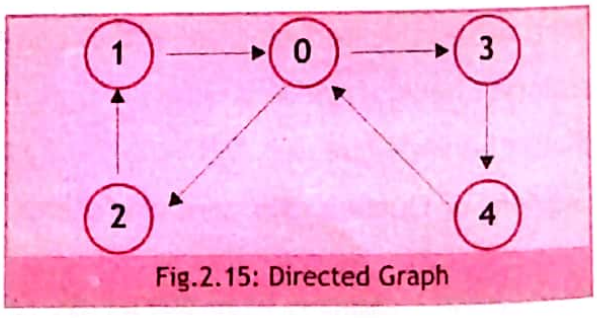


Fig.2.15: Directed Graph

Primarily, there are two types of graphs:

Connected Graph: A graph where at least one path exists between every pair of node. If you can start at any node and reach any other node (in any direction), it is connected. An example is shown in Fig.2.14.

Directed Graph: A graph where edges have direction associated with them as represented in Fig.2.15.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

2.1.6 Trees

The trees represent hierarchical structures consisting of nodes. The top most node is called root. The directly connected nodes make a relationship of parent-child. The higher node in hierarchy is parent, whereas, lower is child. The node which do not have child is referred as leaf node. The relationship between nodes is referred as edge. Each node consist of two things: value and pointer to its child. An example is shown in Fig.2.16.

The trees are crucial for understanding complex relationships between data. Trees are used in various algorithms such as traversal (in-order, pre-order, post-order) and are foundational for understanding more advanced data structures.

In computer Science, Directory structure in an operating system or Tag structure in HTML Language are best examples to understand trees.

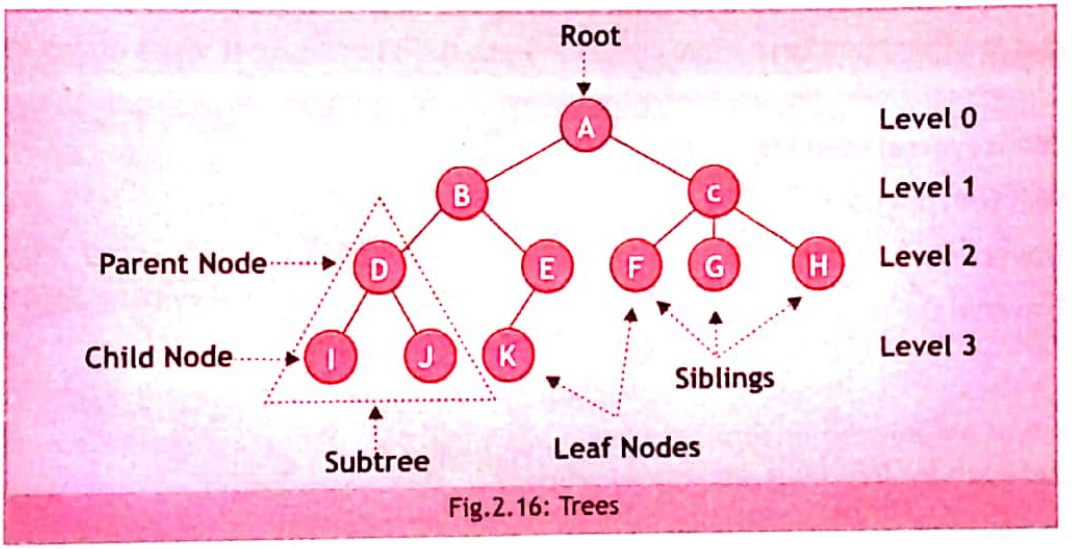


Fig.2.16: Trees

Following are the basic operation that can be performed on trees:

- Insertion: Add a new node at specified location in the tree
- Deletion: Remove existing node
- Search: Find specific node in the tree
- Traversal: Visiting tree in specific order

Tree traversal

A tree traversal is always done in a systematic way and it involves a mechanism to visit all the nodes. There are several methods for tree taversal and each have different purpose. Following is a brief overview of some common tree traversal methods:

- **In-Order Traversal (for Binary Trees) - see Fig.2.17:**
  1. Traverse the left subtree.
  2. Visit the root node.
  3. Traverse the right subtree.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

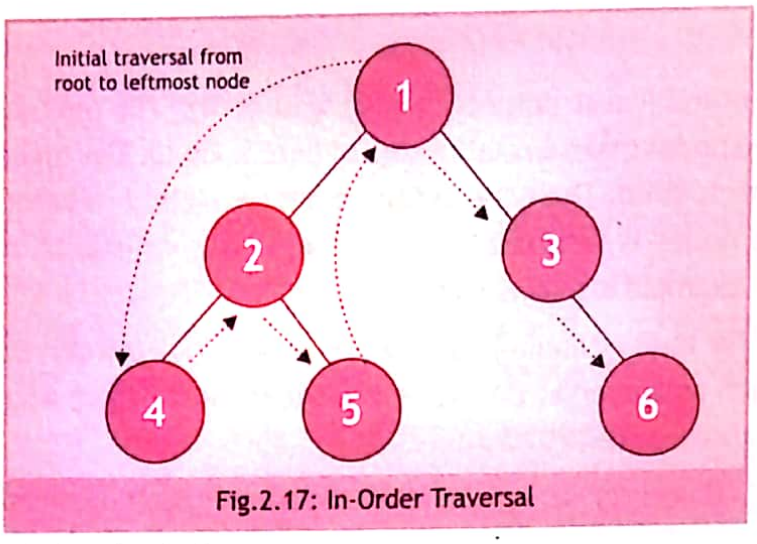


Fig.2.17: In-Order Traversal

So the order of traversal of nodes is 4 -> 2 -> 5 -> 1 -> 3 -> 6.

This traversal is often used with binary search trees (BSTs) because it visits nodes in ascending order.

- **Pre-Order Traversal - see Fig.2.18:**
  1. Visit the root node.
  2. Traverse the left subtree.
  3. Traverse the right subtree.

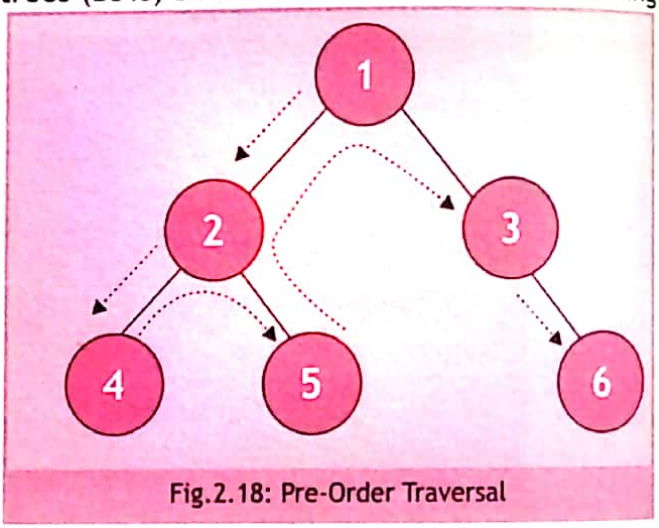


Fig.2.18: Pre-Order Traversal

So the order of traversal of nodes is 1 -> 2 -> 4 -> 5 -> 3 -> 6.

This is useful when we need to work with the root node before its children.

- **Post-Order Traversal - see Fig.2.19:**
  1. Traverse the left subtree.
  2. Traverse the right subtree.
  3. Visit the root node.

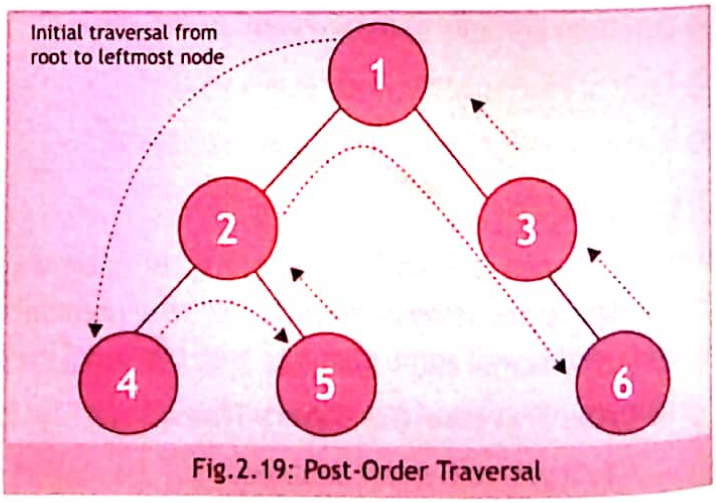


Fig.2.19: Post-Order Traversal

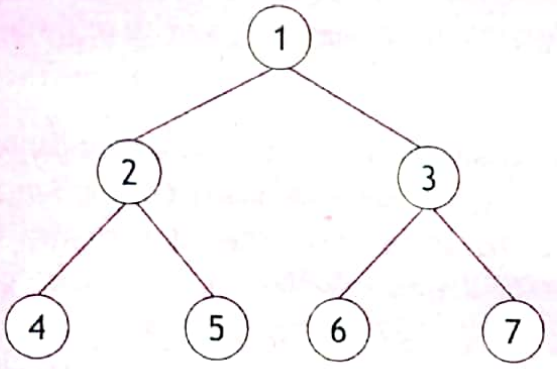
So the order of traversal of nodes is 4 -> 5 -> 2 -> 6 -> 3 -> 1.

This is often used for deleting nodes or freeing memory, as it processes children before their parent.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Example

Fig.2.20 shows examples of Inorder, Preorder and Postorder Traversal



Inorder Traversal

4	2	5	1	6	3	7

Preorder Traversal

1	2	4	5	3	6	7

Postorder Traversal

4	5	2	6	7	3	1

Fig.2.20: Inorder, Preorder and Postorder Traversal

Data Structures Support Computational Thinking:

- **Problem Decomposition:** Different data structures help break down problems into manageable components, making it easier to design and implement algorithms.
- **Efficiency Considerations:** They illustrate trade-offs between different operations (e.g., access time vs. insertion time), helping in choosing the most appropriate structure for a given problem.
- **Algorithmic Thinking:** Understanding these structures aids in developing and analyzing algorithms, improving problem-solving skills by selecting the right tool for each task.

Overall, data structures are not just tools for storing data; they shape the way we think about solving problems and designing algorithms. Mastery of these concepts is crucial for effective computational thinking and developing efficient algorithms.

2.2 Evaluating Computational Solutions

When evaluating computational solutions, it is essential to consider the following criteria:

2.2.1 Correctness

An algorithm is considered correct if it generates the expected output for all valid inputs. To verify correctness, multiple test cases should be used that cover various scenarios, including edge cases.

Example 1: Sorting Algorithm using Bubble Sort Algorithm

Test Case: Given an unsorted array [5, 2, 9, 1, 5, 6], Bubble Sort should correctly sort to [1, 2, 5, 5, 6, 9].

If the algorithm consistently produces the sorted array for various test cases, including edge cases like an empty array or an array with one element, it can be considered correct.

Example 2: Search Algorithm using Binary Search Algorithm

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Case: Given a sorted array [1, 2, 3, 4, 5] and a search key 3, Binary Search should return the index 2 (assuming zero-based indexing).

To verify correctness, ensure Binary Search works for various search keys (existing and non-existing) and for arrays of different sizes.

2.2.2 Clarity

An algorithm is clear if its steps are logically structured and easily understandable. For this purpose, multiple things should be taken care of, for example using descriptive names for variables and functions and ensuring that the steps are ordered logically. Similarly, documentation and comments in code can also enhance clarity.

Example: Consider two python implementations of a function that calculates the factorial of a number.

Case 1: Clear Code

python

```python
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

```

Case 2: Less Clear Code

python

```python
def fct(n):
    return 1 if n == 0 else n * fct(n - 1)

```

The first case is easily readable and understandable because the function name (factorial) represents what logic inside the function will be doing. Similarly, the conditional statement (if n == 0) is also very clear that what it means in the code. However, the second case uses a shorter function name and a ternary operator, which, while valid, may be less immediately understandable to someone new to the code.

2.2.3 Efficiency

Efficiency refers to how well an algorithm uses computer resources, such as time and memory. It is

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

often expressed using Big O notations (e.g., O(n), O(log n), O(n^2)).

Time Complexity:

Time complexity provides a way to evaluate efficiency of a solution. This is done by estimating the time required to solve a problem. While solving a problem, the main outcome of the solution is algorithm. Therefore, this measures the execution time of an algorithm with respect to the size of input. The input size refers to problem size and it represents the size of task and amount of data.

Space Complexity:

Like Time complexity, it evaluates the efficiency of solution but focus on memory used by the solution. This measures the amount of memory an algorithm uses with respect to the size of input.

When evaluating the efficiency of an algorithm, we often focus on the worst-case scenario. For instance, we are searching for a specific element in an array by comparing it with each element at every index, the worst-case situation occurs when the desired element is at the very last index. Big O notation helps us understand the upper bound of an algorithm's running time or space usage relative to the size of the input, providing a way to gauge its efficiency in the most challenging conditions.

Common Big O Notations:

- **O(1) - Constant Time:** The algorithm's running time is constant, regardless of input size. For example, finding a given member in a list involves the same length of time regardless the list has 10 or 10,000 entries.
- **O(logn) - Logarithmic Time:** Algorithm execution time gradually increases with increasing input size. This is comparable to looking for a word in a dictionary, in which each step reduces the search space by half.
- **O(n) - Linear Time:** The algorithm's execution time scales up with input size i.e. if an input size doubles, so does the processing time. One example is a basic loop that goes through each entry in a list.
- **O(n log n) - Linearithmic Time:** This time-based complexity combines linear and logarithmic growth. It is found in effective sorting algorithms such as Merge Sort and Quick Sort, in which the algorithm separates the problem into smaller sections and processes each one separately.
- **O(n^2) - Quadratic Time:** Execution time increases dramatically with input size. This is characteristic of algorithms that use nested loops and compare every member in a list to every other part. When the input size is doubled, the time required to finish the algorithm quadruples.

Example 1: Binary Search Algorithm

Binary search is a search algorithm that locates a target value in a sorted array. It operates by repeatedly partitioning the search interval.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Algorithm Steps:

1. Start by examining the number in the center of the list.
2. Check to see whether this middle number equals the one we are looking for.
3. If it matches, we have located the correct number.
4. If the number we are searching for is on the left part of the list. Repeat the process while ignoring the right side and focusing on the left.
5. If the number we are looking to get is in the list's right half. Ignore the left side and focus on the right portion, continuing the procedure again.
6. Carry on to filter across the list till we identify the number or figure out that it is not on the list.

Time Complexity

The Binary Search Algorithm has Time Complexity of O(log n). The binary search divides the search space in half with each step, this halving process means that the number of elements to search through is reduced exponentially.

When the algorithm starts its initial state is an array of n elements. At each step, we apply the divide and conquer process. This is done by comparing the target value to the middle element of the array or current search range. This comparison helps in eliminating half of the current search range (this half could be either half of the current search range.

- After the first step, we have left with n/2 elements.
- After the second step, we have left with n/4 elements.
- After k steps, we have left with n/2k elements.

The process continues until the number of remaining elements is 1 (or the target is found). The number of steps required is relative to the logarithm of n because we keep halving the problem size each time. Specifically, the number of steps is log2(n).

Space Complexity

The Binary Search Algorithm has the space complexity of O(logn) - Logarithmic Space. To track the current state of recursion, the binary search dependes upon the stack. Each recursive step adds a new frame to the stack. For worst case scenario, the number of the recursions is relative to the number of times the problem size can be halved, which is log2(n). Thus, the space complexity is logarithmic due to the recursive stack usage.

Example 2: Linear Search Algorithm

For Linear search, each element in the array is examined one by one till the target value is found or we reach at the end of the array.

Algorithm Steps:

1. Start at the beginning of the array and examine each element one by one.
2. Compare target value to each element of the array .

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3. The search is completed when the target value matches with an element.

4. If the array ends and no match is found, conclude that the target is not in the array.

Time Complexity

The Linear Search Algorithm has the Time Complexity of O(n), because the required number of operations increase linearly with the size of the input. When the algorithm starts its initial state is an array of n elements. We check each element in sequence e.g. First element, second element and so on up to the nth element. For worst case scenario, every single element in the array is examined before finding the target value (or concluding that it's not in the array).

Space Complexity

The Linear Search Algorithm has the Space Complexity of O(1) - Constant Space. The Linear search utilizes fixed amount of extra space regardless of the size of the input. It only needs a few variables to store the current index and the target value, which does not scale with the size of the input. Thus, its space complexity is constant.

2.2.4 Case Studies

Problem 1: Find Greatest Common Divisor (GCD)

Problem Statement: Given two integers, find their greatest common divisor (GCD).

Solution: Euclidean Algorithm

Algorithm:

1. Take two integers a and b where a ≥ b.
2. While b is not zero:
   - Set a to b
   - b to a % b (the remainder of a divided by b).
3. When b becomes zero, a contains the GCD.

Efficiency:

- Time Complexity: O(log min(a,b))
- Space Complexity: O(1) as only a few variables are used.

Clarity:

- The Euclidean Algorithm is elegant and clear, involving a simple iterative process to find the GCD.

Correctness:

- The algorithm correctly finds the GCD based on the mathematical properties of division and remainders.

Problem 2: Count Numbers of Vowels in a String

Problem Statement: For an input string, count the number of vowels in that string.

Solution: Simple Iteration

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Algorithm:

1. Initialize a counter to zero.
2. Iterate through each character in the string.
3. For each character, check if it is a vowel (i.e., one of 'a', 'e', 'i', 'o', 'u').
4. If it is a vowel, increment the counter.

After the iteration, the counter will contain the number of vowels.

Efficiency:

Time Complexity: O(n), where n is the length of the string.

Space Complexity: O(1) as only a counter variable is used.

Clarity:

The algorithm is simple and clear, involving basic iteration and conditional checks.

Correctness:

This method correctly counts the number of vowels by evaluating each character.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Decomposition:** Breaking down the complex problem into smaller parts, making it easier to manage and solve. For example, building a website involves separate tasks like designing, coding, backend setup, and testing.
- **Pattern Recognition:** Identifying commonalities or trends within a problem or across similar problems. For instance, in data analysis, recognizing user behavior patterns can help enhance product design.
- **Abstraction:** Focusing on the essential details of a problem while ignoring unnecessary ones, simplifying complex systems. For example, creating a user interface mockup focuses on key interactions without detailing all technical aspects.
- **Algorithm Design:** Creating a clear, step-by-step process to solve a problem efficiently, which is essential in programming. For example, sorting numbers involves using a defined procedure to arrange them from smallest to largest.
- **Correct:** Produces the desired result for all valid inputs.
- **Clear:** Easy to understand and follow.
- **Efficient:** Optimizes resources like memory and time.
- **Sorting Algorithms:** Organize data in a particular order (e.g., Quick Sort, Bubble Sort).
- **Searching Algorithms:** Find specific items in data (e.g., Linear Search, Binary Search).
- **Graph Algorithms:** Work with nodes and edges (e.g., Dijkstra's Algorithm, Depth-First Search).
- **Dynamic Programming:** Solves problems by breaking them into smaller subproblems and storing solutions (e.g., Fibonacci numbers).
- **Greedy Algorithms:** Make the best choice at each step for overall optimization (e.g., Huffman Coding).
- **Backtracking Algorithms:** Explore solutions and backtrack when necessary (e.g., N-Queens Problem, Sudoku Solver).
- **Binary Search:** Efficient for finding a target value in a sorted array, with a time complexity of O(log n).
- **Linear Search:** Examines each element sequentially in an unsorted array, with a time complexity of O(n).
- **Lists and Arrays:** Store elements in sequence, allowing random access via indices, and are crucial for operations involving iteration and loops.
- **Stacks:** Follow the Last-In-First-Out (LIFO) principle, commonly used in algorithms like depth-first search.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. Which of the following best defines a data structure?
   a. A method of communication
   b. A way to store and organize data in memory
   c. A system for internet access
   d. A graphical interface design
2. Which data structure allows dynamic memory allocation and easy insertion/deletion?
   a. Array
   b. Stack
   c. Linked List
   d. Queue
3. In a singly linked list, each node contains:
   a. Data and address of previous node
   b. Only data
   c. Data and address of next node
   d. Index value and data
4. The main advantage of a doubly linked list over a singly linked list is:
   a. Random access
   b. Uses less memory
   c. Allows traversal in both directions
   d. Faster arithmetic operations
5. What type of linked list connects the last node back to the first?
   a. Doubly linked list
   b. Circular linked list
   c. Singly linked list
   d. Linear linked list
6. Which operation removes the top element from a stack?
   a. Enqueue
   b. Dequeue
   c. Pop
   d. Top
7. Which node in a tree does not have any children?
   a. Root node
   b. Leaf node
   c. Sibling node
   d. Parent node
8. Which data structure operates on a Last-In, First-Out (LIFO) principle?
   a. Array
   b. Stack
   c. Queue
   d. Tree
9. Which algorithm is used for sorting data?
   a. Quick Sort
   b. Binary Search
   c. Depth-First Search
   d. Euclidean Algorithm
10. Which algorithm is an example of a Greedy Algorithm?
    a. Bubble Sort
    b. Huffman Coding
    c. Depth-First Search
    d. Binary Search
11. What is the space complexity of the Binary Search algorithm?
    a. O(1)
    b. O(n)
    c. O(log n)
    d. O(n^2)
12. Which data structure allows random access to elements by index?
    a. Stack
    b. Queue
    c. Array
    d. Tree
13. A school wants to automate the process of generating student report cards based on input marks. Which step of computational thinking would most likely involve identifying how to separate marks by subject, calculate averages, and assign grades?
    a. Decomposition
    b. Pattern Recognition
    c. Abstraction
    d. Algorithm Design

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

14. What is the time complexity of Linear Search?

a. O(1) b. O(log n) c. O(n) d. O(n^2)

15. During an inter-school tech competition, students were asked to develop a solution for managing traffic signals based on traffic density. Which approach best shows their use of computational thinking?

a. They memorized how traffic lights work in real life.

b. They created a flowchart outlining how traffic signal timing changes with vehicle count.

c. They guessed a solution based on their intuition.

d. They programmed a random timer for each signal.

Give short answers to the following Short Response Questions (SRQs).

Problem 1: Finding the Sum of Digits

Problem Statement: Given an integer, calculate the sum of its digits.

Solution: Digit Extraction and Summation

Give answer of the following:

a) Algorithm

b) Efficiency

- Time Complexity
- Space Complexity

c) Clarity

d) Correctness

Problem 2: Finding the Factorial of a Number

Problem Statement: Compute the factorial of a non-negative integer n (i.e., n!).

Solution: Iterative Approach

Give answer of the following:

a) Algorithm

b) Efficiency

- Time Complexity
- Space Complexity

c) Clarity

d) Correctness

Problem 3: Finding the Largest Number in a List

Problem Statement: Given a list of integers, find the largest number in the list.

Solution: Linear Scan

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Give answer of the following:

a) Algorithm

b) Efficiency

- Time Complexity
- Space Complexity

c) Clarity
d) Correctness

Problem 4: Checking for Prime Numbers

Problem Statement: Determine if a given integer n is a prime number (i.e., it has no divisors other than 1 and itself).

Solution: Simple Divisibility Test

Give answer of the following:

a) Algorithm

b) Efficiency

- Time Complexity
- Space Complexity

c) Clarity
d) Correctness

Problem 5: Finding the Average of a List

Problem Statement: Calculate the average value of a list of numbers.

Solution: Summation and Division

Give answer of the following:

a) Algorithm

b) Efficiency

- Time Complexity
- Space Complexity

c) Clarity
d) Correctness

6. How can abstraction help a student create a simple timetable app for school?
7. A shopkeeper wants to use a program to quickly find a product in a long list. Which type of algorithm should they use and why?

Give long answers to the following Extended Response Questions (ERQs).

1. Compare and evaluate the efficiency of Bubble Sort and Merge Sort in sorting a list of 1,000 student names. Discuss in terms of time complexity and clarity. Which is more suitable and why?
2. Describe the difference between Stack and Queue. How does their operation affect algorithm design? Provide one real-life example for each data structure.
3. Explain the importance of tree traversal techniques. Compare in-order, pre-order, and post-order traversal with examples and their use cases.

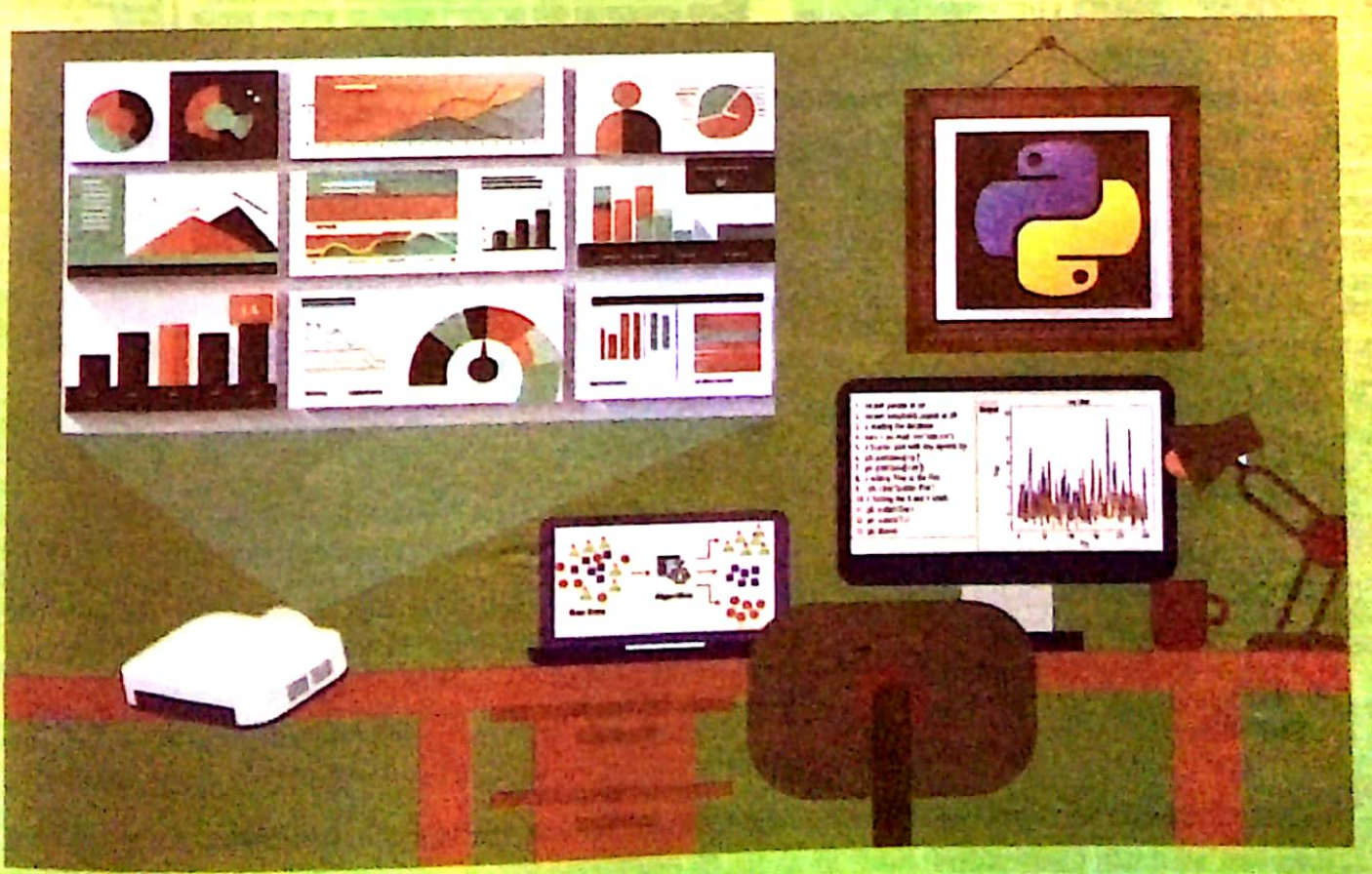
Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3 Programming Fundamentals

Learning Outcomes

At the end of this unit students will be able to:

- understand and evaluate applications of various programming paradigms.
- use more advanced programming constructs such as data structures (lists etc.), file handling (disk IO to write to storage), and databases in Python.
- implement complex algorithms that use lists etc. in Python
- determine more advanced techniques (unit tests, breakpoints, watches) for testing and debugging their code in Python



Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.1 The Programming Paradigm

The term "programming paradigm" describes a vital process for creating and organizing computer program code. It helps in the structure of code, program flow and managing of data for programmers. In other words it can be termed as a particular methodology to use programming language for problem solving. Although programmers can design their own unique structures and functions inside a language, these can't be termed as a paradigm, as it is the programming language that specifies and provides guidelines and functions in terms of what and how to develop. Thus, a programming language paradigm offers a particular way to arrange various functionalities in order to accomplish specified objectives. Additionally, it facilitates the developers with standard tools and implementation techniques that are used commonly for effective problem-solving which helps developers to focus on the specific task in hand and design solution effectively. The most commonly used programming languages include Python, Java, SQL, etc.

3.1.1 Functional Programming

Functional programming is the paradigm of software development which was and is followed by many, especially for small applications where the code is divided into functions based on functionality. The developers in functional programming focus more on 'what' is to be achieved rather than 'how' to address the problem. The solution of the problem in hand is divided into segments and each segment is typically addressed in functions. This way, every step of the solution is mapped in a group of functions as shown in Fig. 3.1.

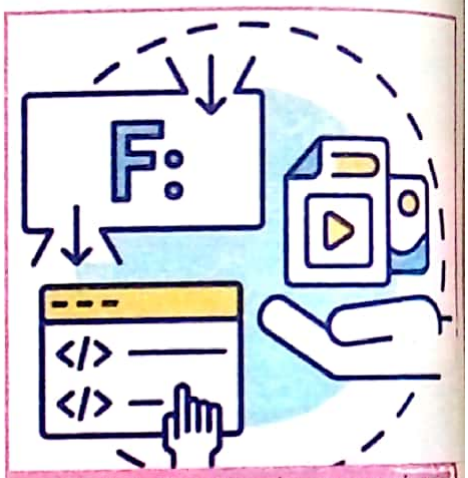


Fig.3.1: Functional programming

Table.3.1: Pros & Cons of Functional Programming

Pros	Cons
Simpler code.	Not easy for beginners. Needs practice
Code is easier to track for logic and debug.	May involve with overhead especially when handling large data structures.
Functions are easier to test and run.	Complex algorithms may be harder to express, often requiring higher-order functions.
Naturally supports parallel and concurrent programming.	Requires careful handling of data sharing and synchronization for efficient execution.
Emphasizes on what to compute rather than how, hence more concise code.	May be less efficient in performance-critical scenarios.

So, if the flow is clear how to achieve the solution, it becomes easier in functional programming. This way, updating the code for any changes becomes simpler as only that particular functions(s need to be modified and whole program does not need to be read and understood

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.1.2 Object Oriented Programming

Object Oriented Programming (OOP) is software development paradigm based on the data and objects, which include functions (methods) that define the behavior and data (attributes) to indicate the properties of objects. These are the program's building blocks to regenerate the actual objects and their functionalities, in code. The programs designed using OOP paradigm represents the physical attributes of the objects from real world and therefore developers have a more natural understanding of the problem to be solved by devising the program around objects and their actions.

The main advantage of using OOP paradigm is that the emphasis in problem solving is on objects and not the logic. The program reflects real-world objects like car, fruit, etc. This approach makes it easier to understand the problem, design solution and change the code if needed. OOP is implemented through classes which allow defining initially the object and their functionalities and later physical instances are created on spot when need to be used in the program. This is quite useful for complex systems where developers can map objects and their associations to be realistic and handle them more efficiently. OOP limits direct access to data for safety and reliability of data and programs.

Table.3.2: Pros & Cons of Object Oriented Programming

Pros	Cons
Simplifies complex systems by breaking them into smaller reusable components.	May result in complex designs if not implemented thoughtfully.
Hides implementation details, hence improving code maintainability and security.	At times code is harder to understand and debug due to encapsulation.
Establishes hierarchical relationships between classes.	Rigid design due to inheritance may reduce flexibility.
Objects of different types are treated as if they were of the same type.	Polymorphism increases complexity, making code harder to understand and reason about.
Represents real-world objects and their relationships effectively.	May not be well-suited for certain problems, such as computationally intensive tasks.

Key Concepts

A **class** is an object creation stencil or template. It specifies a collection of methods (functions) and attributes (variables or properties) that its objects will hold. You are creating an instance of a class when you create an object from it. Although each object has a distinct set of data (attribute values), they all use the same class-defined methods. You may effectively express and control complicated data structures and behaviors in your program with this paradigm.

An **object** is an instance of a class that denotes a physical thing, such a person, a car or a bank account. Every object has distinct properties (attributes) and behaviors (methods). For instance, each home (object) constructed from a class that specifies the structure of a house has particular

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

properties, such as size, color or number of rooms..All objects have the same basic structure as defined by the class, apart its own specific characteristics. Objects in a program interact with one another to imitate real-world situations and improve the organization and modularity of the code.

The 'StationaryItem' class (as shown in table) acts as the stencil for all stationary items in the bookstore. Individual goods developed by this session include pens, pencils and notebooks. Each item has unique values for attributes such as name, price and color. The methods of the class allow you to retrieve or change these attributes which allows managing of information about various stationary items easily.

Class: StationaryItem		
Attributes:	Object 1: A blue XYZ pen.	Object 2: A red ABC pencil.
name (e.g., "Pen", "Pencil", "Notebook")	name: "Pen"	name: "Pencil"
price	price: 200	price: 100
color (e.g., "Blue", "Black", "Red", "Green")	color: "Blue"	color: "Red"
brand (e.g., "XYZ", "ABC", "DEF")	brand: "XYZ"	brand: "ABC"
It can have methods like get_price() to obtain the price of the item, set_price() to adjust the price of the said item, etc.		

Encapsulation is the process of information hiding by combining data (attributes) and the methods that operate on the data into a single entity, i.e. a class. The main concept is to keep together the data and methods that are relevant. Encapsulation does not allow direct access to some components but via some designated methods provided by the object. This way internal working of an object is hidden. Encapsulation protects data from unwanted access or changes and also improves code to be written in modules and easy to read and understand.

For instance, in a bank, the account holder's name, account number and balance are critical information. Rather than changing the balance directly, deposit(amount) method will add the amount to the balance while withdraw(amount) method will subtract the said amount from the balance. Further, get_balance() method will retrieve current balance without changing the balance figures. This way, critical information are encapsulated and cannot be modified directly.

Abstraction simplifies creating complicated scenarios by showing only the necessary information and usefulness. It provides only essential information about the data to others, hiding the background details or implementation. Consider a vehicle for example, where we are mainly concerned with common operations such as start, stop and accelerate. But the underlying mechanism of how engine burns fuel and shifting of gears, etc. are kept hidden.

Inheritance allows you to create new classes (subclasses or derived classes) from existing parent classes. Subclasses inherit the attributes and methods of their parent class, helping in reducing repetition of codes. This idea promotes the efficient extension of existing structures. In general

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

terms, a 'Vehicle' may have attributes such as speed, color, wheels and operations like accelerate() and brake(). If we specify it further for a 'Car' which is a type of 'Vehicle', so we do not need to define the above stated attributes and operations again because 'Car' inherited these from 'Vehicle' being a derived/sub class and automatically owns these properties and functions of 'Vehicle'. However, 'Car' may have its own additional properties like 'seatingCapacity' and operation like openSunroof().

Polymorphism allows objects from various classes to be considered as instances of a common type, though the implementation of the common type differs among the various classes in their own unique way. This allows writing generic code that can deal with a wide variety of objects. Polymorphism makes it easier to modify object behavior.

For example, the common type 'Shape' can be a triangle, rectangle or circle. Though all three are shapes but have their own distinct properties and operations like draw, resize, calculate-area, etc. Polymorphism is useful in such scenarios where a common operation applies for all but implementation differs based on the type of shape. In context of specific shapes, triangle needs base and height, rectangle uses length and width while circle requires radius to calculate the area. Instead of writing three separate specific calculate-area methods (which may increase the more shapes we take into consideration), via polymorphism we can send a simple instruction to different objects such as 'draw-yourself' and each object will perform the operation in its own prescribed way on the basis of type of shape.

3.2 Programming Constructs in Python

3.2.1 List

List is a basic data structure in Python that holds groups of elements in a sequential manner. This flexibility allows list to store elements of various data types such as boolean, integers, etc. The order that things are added is maintained by lists. Index 0 will have the first item inserted, index 1 will have the second, and so on. This arrangement is essential for retrieving elements based on their location. Different data types can be stored as elements in the same list. List sizes are dynamic, i.e. elements can be added or removed as required.

In Fig.3.2, let's utilize the common functions of adding and removing items from the list. Initially, on line 1 we defined a list namely 'student_grades' with 5 elements in it, to store grades of a student for instance. On line 3, we inserted an additional grade of 90 in the list on index 2, using the **insert()** function. Alternative to insert any other value is by using the **append()** function, which we used in line 5 and inserted another value of 100 which was added at the end of the list. Thereafter, on line 8 the value of 78 from the list was deleted using **remove()** function. Another way to remove any element from the list is using the **pop()** function, such that index number needs to be mentioned. This way, we can remove an element from the list either if the element or its index number is known.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
1  student_grades = [85, 92, 78, 95, 82]
2
3  student_grades.insert(2, 90)
4  print("Grades after adding a new grade:", student_grades)
5  student_grades.append(100)
6  print("Grades after adding another new grade:", student_grades)
7
8  student_grades.remove(78)
9  print("Grades after removing a grade:", student_grades)
10 lowest_grade = student_grades.pop(2)
11 print("Grades after removing the grade on index 2:", student_grades)
12
13 if 88 in student_grades:
14     student_grades.remove(88)
15 else:
16     print("Grades88 not found in the list")
17
18 average_grade = sum(student_grades) / len(student_grades)
19 print("Average grade:", {average_grade})
```

Output

```
Grades after adding a new grade: [85, 92, 90, 78, 95, 82]
Grades after adding another new grade: [85, 92, 90, 78, 95, 82, 100]
Grades after removing a grade: [85, 92, 90, 95, 82, 100]
Grades after removing the grade on index 2: [85, 92, 95, 82, 100]
Grade 88 not found in the list
Average grade: {90.8}
```

Fig.3.2: Add and remove items from a list

Activity 1

Extend the program of Fig.3.2 to search for all odd numbers in the list and remove them.

Moreover, there are scenarios, where a particular value is not in the list but you intend to remove it and results in an error. So, a good programming practice is to use conditional (if-else) statement as we have used in lines 13-16. This way, if the number is in the list, it will be deleted otherwise it won't abnormally terminate the program rather a message is printed that the said number is not found, or something like this. Lastly, using the **sum()** and **len()** functions, we calculated the average grade of the student without using any loop.

3.2.2 Tuples

In Python, tuples are represented using '(_,_,_)' and are a special type of list having one feature that they are immutable i.e. a tuple's contents cannot be updated once it has been created. We can view the objects and their arrangements but we cannot add, remove or rearrange them. Tuples are used when a set of values must remain in order and unchanging throughout the program. For example the days of the week (("Monday", "Tuesday", "Wednesday")) or geographical coordinates (latitude and longitude), etc.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
1 grades = (85, 92, 78, 92, 85, 95)
2 print(grades)
3 print("First grade:", grades[0])
4 print("Third grade:", grades[2])
5 if 92 in grades:
6   print("Student got 92")
7 print(grades.count(92))
8 new_grades = grades + (88, 90)
9 print(new_grades)
```

Output

```
(85, 92, 78, 92, 85, 95)
First grade: 85
Third grade: 78
Student got 92
2
(85, 92, 78, 92, 85, 95, 88. 90)
```

Fig.3.3: Tuples

On line 1 we create a tuple of student grades namely 'grades' and initialize it with some sample grades and print it in the next line which shows duplicate values as well. However, individual grades are accessed by specifying the index (which starts from 0) on line 3 and 4. To check if a specific grade exists in the tuple the `in` operator can also be used with tuples as we used in line 5. Additionally, to find the number of occurrences of a grade `count()` method returns the number of times a specific value appears in the tuple as shown in line 7. Line 8 shows that we can create a new tuple by concatenating existing tuples using the `+` operator.

3.2.3 Sets

Sets in Python are represented using '{_,_,_}' and are collections of unique and unordered elements i.e. a set cannot comprise of duplicate values. A set will only hold one instance even if the same value is added to it more than once. Sets can be used to store a list of unique usernames, product IDs, etc. Using the 'in' operator it is quite easy to check whether a particular element exists in a set or not.

Initially on line 1 a set of student grades is created namely 'grades' and is initialized with some sample grades. Notice that duplicate grades (85 and 92) exist in the initial assigned values. Since sets only store unique values, the duplicates will be automatically removed. Line 2 prints the set 'grades', demonstrating that the duplicates have been removed.

Next, we add a new grade to the set i.e. 88 on line 3 using the `add()` method. Line 4 prints the set again showing the updated values. To check if a specific grade exists in the set, the `in` operator is used. In line 5, if 92 is found in the set, then a message is printed.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
1 grades = {85, 92, 78, 92, 85, 95}
2 print(grades)
3 grades.add(88)
4 print(grades)
5 if 92 in grades:
6   print("student has 92 among the grades.")
```

Output

```
{92, 85, 78, 95}
{78, 85, 88, 92, 95}
Student has 92 among the grades.
```

Fig.3.4: Sets

3.2.4 Dictionary

Dictionary Store Key-Value Pairs

Dictionary is an essential data structure in Python that is used to store and arrange data in an organized way using key-value pairs such that each key represents a distinct value, i.e. each key is associated with a particular value. List have index while in dictionary it can be of any data type. For example, in a phone book where the phone number (value) and the person's name (key) are associated with each other. By searching for the matching name, we may quickly get the phone number. Because of this fundamental idea, dictionaries may be used for storing and retrieving information by associating unique keys with values.

In order for keys to be used for searching, they must be unchangeable types of data like strings, integers or tuples. Values may store complicated data structures with flexibility since they can be any data type including texts, integers, lists and even other dictionaries. By design dictionaries avoid duplicate keys such that every key acts as a unique label hence avoiding reshuffling of data. Dictionaries maintain an ordered link among keys and their corresponding values, like keeping track of passwords and usernames, etc.

The Python code in Fig.3.5 shows how to add, create and retrieve entries from a dictionary. Initially a dictionary is initialized with 3 student records where student names being the key and their respective marks are the values associated with the keys. Immediately after that on line 7 and 8 using for loop, we print the dictionary. On line 10, we add another student 'Zaid' in the dictionary, which is added at the end of the list. A similar statement for 'Bilal' is written, since it is already in the dictionary, its value will be updated. On line 13, we removed 'Dilawar' from the dictionary. For confirmation, in the next 4 lines using conditional statements (if-else) we checked for 'Dilawar' in the dictionary. Lastly, final dictionary is printed again. Point to note here is that if we comment line 13, record of 'Dilawar' will not be deleted. The corresponding outputs of the said program and after commenting line 13, both are shown for better understanding.

Next, we modify the code such that an empty dictionary is created and thereafter, name o

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

students and their respective test numbers from the user are taken in, as shown in Fig.3.6. Lastly, the student name is asked to be searched in the dictionary and corresponding search result is displayed.

```
1  student_grades = {
2      'Amir': 90,
3      'Bilal': 85,
4      'Dilawar': 92
5  }
6
7  for student, grade in student_grades.items():
8      print(f"{student}: {grade}")
9
10 student_grades['Zaid'] = 88
11 student_grades['Bilal'] = 87
12
13 del student_grades['Dilawar']
14
15 if 'Dilawar' in student_grades:
16     print(f"\n Dilawar's grade:", student_grades['Dilawar'], "\n")
17 else:
18     print("Student Dilawar not found \n")
19
20 for student, grade in student_grades.items():
21     print(f"{student}: {grade}")
```

Output

output without line 13:

```
Amir: 90
Bilal: 85
Dilawar: 92

 Dilawar's grade: 92

Amir: 90
Bilal: 87
Dilawar: 92
Zaid: 88
```

Confirmation via printing

Output with line 13:

```
Amir: 90
Bilal: 85
Dilawar: 92
Student Dilawar not found

Amir: 90
Bilal: 87
Zaid: 88
```

Confirmation via printing

Fig.3.5: Program to create dictionary for storing students' grades

Activity 2

Extend the program of Fig 3.5 in such a way that if the searched item is not found in the dictionary, ask the user if it needs to be added. If user presses 'y' for yes, then the key and respective value (to be taken input) are to be added.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
student_test_number = {}
while True:
    name = input("Enter student name (or 'q' to quit): ")
    if name.lower() == 'q':
        break
    grade = float(input("Enter Test-Number for {}: ".format(name)))
    student_test_number[name] = grade

name = input("Enter student name to view test-number: ")
if name in student_test_number:
    print(f"{name}'s test-number: {student_test_number[name]}")
else:
    print(f"Student '{name}' not found.")
```

Output

```
Enter student name (or 'q' to quit): Amir
Enter Test-Number for Amir: 90
Enter Student name (or 'q' to quit): Bilal
Enter Test-Number for Bilal: 87
Enter Student name (or 'q' to quit): Dilawar
Enter Test-Number for Dilawar: 92
Enter student name (or 'q' to quit): Zaid
Enter Test-Number for Zaid: 88
Enter student name (or 'q' to quit): q
Enter student name to view test-number: Bilal
Bilal's test-number: 87.0
```

Fig.3.6: Search an entry in dictionary

Note that, it takes significantly less time to find a value in a dictionary than it does to search a list in Python. This is the reason because in lists we need to go through each item individually in order to locate what you're searching for whereas in dictionaries using the key-value pair, we can locate the searching item quickly.

Finding a Value in a Dictionary v/s List

For the sake of better understanding, a program was written where a list and dictionary were randomly populated with 1 lakh entries. Such that a list of strings of 4-characters was setup while for every 4-character in the dictionary a corresponding key value was randomly arranged. Next 1 random element was searched in both the programming constructs and their times were noted. The difference of time clearly shows the searching in dictionary is faster as compared to the list, as shown in Fig.3.7. Sample of populated data in dictionary and list are shown in first two lines.

```
Search results (Dictionary): {20145: 'dlei'}
Search results (List): ['uiib']
Dictionary search time: 1.000166 msec
List search time: 3.000259 msec
Difference of time: Dictionary Time - List Time: -2.000093 msec
```

Fig.3.7: Difference of time to Search in dictionary v/s list.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.2.5 Nested List

A nested list is a list of lists such that the inner lists or sub-lists can further contain list as elements. Nested list is quite useful in handling complex data like matrices. So, to explore nested list let's try to create a matrix and print it. First, we define a 3x3 matrix. This is achieved using a nested-list such that every element of the list is itself a list and contains row-wise data. As shown in Fig.3.8, we initialized a 3x3 matrix and thereafter on line 6 just by passing the matrix name as an argument to the print() function, we can get it printed. The first line of the output clearly distinguishes the multiple lists that exist in the list. Next, we print the matrix on line 8 using a for-loop. Since, the list has only 3 members, the loop will execute thrice and in every iteration a row gets printed, hence in the form of a matrix. Lastly, on line 11 we use nested loop, such that the outer loop index is 'row' while the inner loop index is 'col'. This way, for the first row, all columns will be printed and then the same will be processed for second and third rows and the whole matrix gets printed in a single line. To modify the code and get it printed in the form of a matrix, is left as an activity for you.

```
1  matrix = [
2      [1, 2, 3],
3      [4, 5, 6],
4      [7, 8, 9],
5  ]
6  print(matrix)
7
8  for i in matrix:
9      print(i)
10
11 for row in range(len(matrix)):
12     for col in range(len(matrix)):
13         print(matrix[row][col], end=', ')
```

Output

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
1, 2, 3, 4, 5, 6, 7, 8, 9,
```

Fig.3.8: Printing a Matrix using Nested List

In this code, a 3x3 matrix is defined, and its transpose is calculated and printed. In line 2-4, the original matrix is given with numbers. In line 7, the code initializes an empty list to store the transpose. Lines 9-12 iterate through each column of the original matrix and create rows for the transpose. In line 13, each new row is appended to the transpose list. Finally, in lines 16-17, the code prints the transposed matrix. The transpose swaps the rows and columns of the original matrix.

Thereafter, we extend the program to calculate the transpose of the matrix and print it. For this purpose, on line 10 of Fig.3.9 we initialized another matrix. On line 16, just like in the last example, we placed a nested loop as we know how to access the nested list contents one-by-one. But, while assigning these values to the transpose matrix the rows should be converted into columns and vice versa. Therefore, note that the indices of the transpose matrix are reversed, which will result in the element at row 1 and column 2 to be placed at column 1 and row 2, which was the desired outcome. In lines 17-19, the output reflects successful creation of transpose matrix of the given matrix.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Activity 3

Tailor the code of Fig 3.8, such that for all odd values of the matrix the sign should be inverted while for all even values multiply it by 2.

```
matrix = [
     [1, 2, 3],
     [4, 5, 6],
     [7, 8, 9]
]
print('Matrix is: ', matrix)
for i in matrix:
    print(i)

trans = [
     [0,0,0],
     [0,0,0],
     [0,0,0]
     ]

for row in range(len(matrix)):
    for col in range(len(matrix)):
        trans[col][row] = matrix[row][col]

print('Transpose of the Matrix is: ',trans)
for j in trans:
    print(j)
```

Be careful about the indices as it may return the original matrix and not the transpose of the matrix.

```
Matrix is:  [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
Transpose of the Matrix is:  [[1, 4, 7], [2, 5, 8], [3, 6, 9]]
[1, 4, 7]
[2, 5, 8]
[3, 6, 9]
```

Output

```
Matrix is:  [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
Transpose of the Matrix is:  [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
```

Fig.3.9: Program to create Transpose of a matrix

Activity 4

A matrix 'X' is symmetric matrix, if X = (Transpose of X). Write code to determine whether a given matrix is symmetric matrix or not.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.2.6 List as a Value in Dictionary

A dictionary comprises of a key-value pair and for multiple values against a key can be stored in the dictionary in the form of a list. For example, every record of a student can be stored in the dictionary such that the student's name will act as the key while the registration number and marks will be the values. To store multiple values, a list can be used.

```
students = {
    "Sara": ["345", 85, 90, 78],
    "Jamal": ["678", 75, 88, 92],
    "Zaid": ["912", 80, 79, 95]
}

sara_reg_num = students["Sara"][0]
print("Sara's Reg. No.: ",sara_reg_num)
jamal_marks = students["Jamal"][1]
print("Jamal's Marks for first subject are: ",jamal_marks)

students["Azra"] = ["234", 78, 85, 90]
print(students, "\n")

student_name = "Azra"
subject_index = 2
new_mark = 100
if student_name in students:
      students[student_name][subject_index] = new_mark
      print(f"*** Marks updated for {student_name} at index {subject_index} \n")
else:
    print("Student not found \n')

for name, details in students.items():
    print(f"Name: {name}")
    print(f"Registration Number: ", {details[0]})
    print(f"Marks: [", end='')
    print(details[1],',',details[2],',',details[3], "]\n")
```

Output

```
Sara's Reg. No.:  345
Jamal's Marks for first subject are:  75
{'Sara': ['345', 85, 90, 78], 'Jamal': ['678', 75, 88, 92],
 'Zaid': ['912', 80, 79, 95], 'Azra': ['234', 78, 85, 90]}

*** Marks updated for Azra at index 2

Name: Sara
Registration Number:  {'345'}
Marks: [85 , 90 , 78 ]

Name: Jamal
Registration Number:  {'678'}
Marks: [75 , 88 , 92 ]

Name: Zaid
Registration Number:  {'912'}
Marks: [80 , 79 , 95 ]

Name: Azra
Registration Number:  {'234'}
Marks: [78 , 100 , 90 ]
```

Fig.3.10: A dictionary with a list.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Fig 3.10 shows the code how to create a list in a dictionary and initialize it, such that the lis comprises of all the data related to student in every row. Line 7 shows how to extract th registration number from the list while line 9 depicts how to select marks from the list Furthermore, on line 12 we insert a new record in the dictionary and print the whole dictionary i the next line which shows all the records in the dictionary. To understand how to access the lis elements in the dictionary, let's assume that the second subject marks of the record that we jus inserted needs to be updated. In line 19, based on the key being the student's name, we update the marks of the second subject. Lastly, on line 24, using a for loop in terms of the key-value pair we print all the items of the dictionary.

Activity 5

A primary school has subject teachers such that a teacher teaches only 2 subjects to a class. Assuming, each class studies 8 subjects in total. Write a program using list as a value in dictionary, so that 'class' becomes the key and respective 'teachers' are stored as values.

3.2.7 Files in Python

Importance of Disk I/O

Disk input/output (I/O) is vital for efficient data handling and optimal system performance in computers. Disk I/O bridges between the CPU and HDD for information retrieval, such that relevant data is passed to the CPU for processing and later is stored back on permanent storage devices. Almost all computer operations are supported by this continuous data interchange, from launching of the programs, opening files and save changes made by user.

In Python, files are mainly of two types:

Text File:

In a text file, data is stored through characters like .txt, .csv, etc. Point to note here is that a special marker called 'End Of Line' (EOL) generally '\n' is used to highlight completion of a line. This way, Python knows that a new line is starting after EOL marker.

Binary File:

In a binary file data is stored in combination of 0s and 1s, i.e. binary data. Binary files are not meant for humans but rather for other programs to read information. Images (having extension .jpg, .png), audio (.mp3), video (.mp4) and executable programs (.exe) are all examples of binary files.

File Handling Methods and Operations

Data in files from storage devices is brought up in RAM, where functions like creation of new file, reading and updating of already existing file, closing the file, etc. can be performed on the files. Python has file handling methods that let you work with files and you may interact with files in an organized manner and create, read, write and modify their contents programmatically, using:

- The **open()** method is used to access a file such that it takes in 2 arguments, one is file name

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

(with complete path) while the other argument is the desired mode of operation (reading, writing, appending, etc.).

- After the file is opened in read mode, its contents can be extracted by using **read()** function and the complete contents of the file is obtained as a string.
- Write mode allows you to add new material or edit already-existing information to the file using **write()** function.
- You may add new data at the end of an already-existing file without erasing its contents, by using the append mode using the **add()** technique.
- The **close()** function must be used to close a file once you have completed working with it. This guarantees data integrity and releases system memory that was in use by the file.

Example: Program To Create An Empty File And Write Content In It.

For the code in Fig.3.11, in the first line a file named 'abc.txt' is created (if it already does not exist) with **open()** function and returns a file handler. Thereafter, every operation uses this file handler to operate on the file. For example, in the very next line a sentence is written in the file, using **write()** function using 'new_file' file handler. In line 4, a message for the user is displayed such that the contents of the file can be checked by opening it, as shown in the output. Lastly, for every file that is opened, it needs to be explicitly closed before the program terminates, and that is generally the last line of the program.

```
1 new_file = open("abc.txt", "w")
2 new_file.write("The quick brown fox jumps over the lazy dog !!!")
3
4 print("File created and written to successfully!")
5 new_file.close()
```

Output

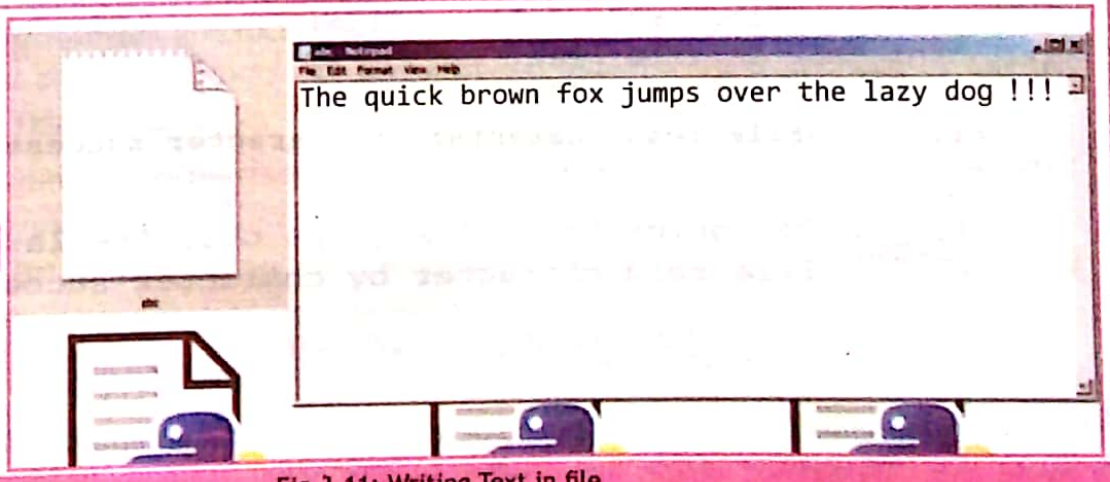


Fig.3.11: Writing Text in file

Alternatively, as shown in Fig.3.12, we can use the 'with' keyword. The benefit of using 'with' keyword for opening file is that the file does not need to be closed explicitly and specially in case of an error, it will automatically be closed. For the sake of clarity, as we are writing the same sentence as of Fig 3.11, but without '!!!' at the end of the sentence, just to distinguish that new text has been updated. When executed, the updated output of the same file is shown in the output.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
1 with open("abc.txt", "w") as file:
2   file.write("The quick brown fox jumps over the lazy dog")
3
4 print("File created and written to successfully!")
```

Output

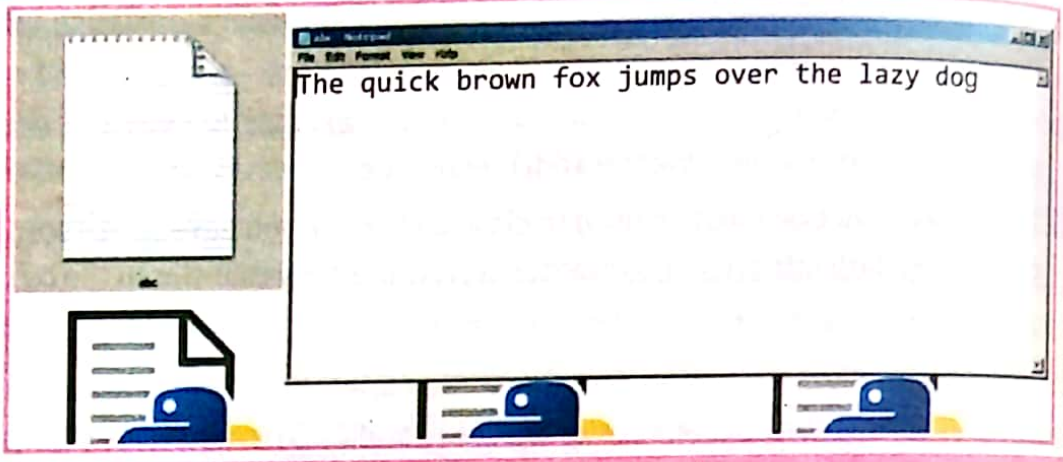


Fig.3.12: Opening file to write with 'with' keyword

Read Existing Files

Next, let's read the contents of an existing file, like the file 'abc.txt' that we just created and saved. The file "abc.txt" is opened in read mode ("r") in the first line of Fig.3.13. An error message will be displayed if the file is missing. As can be seen in line 2, using a for-loop the whole file is read character-by-character and printed on the screen without adding a new line after each character. Lastly, a confirmation message is displayed on line 4 that the file is successfully read, as shown in the output.

```
1 with open("abc.txt", "r") as file:
2   for char in file.read():
3     print(char, end="")
4 print("\nFile read character by character successfully!")
```

Output

```
The quick brown fox jumps over the lazy dog
File read character by character successfully!
```

Fig.3.13: Program to read file

Errors may arise while dealing with files and may result in abnormal termination of the program. To avoid this, 'Try-except' blocks are used to handle possible problems while accessing and working with files. Typical issues include trying to open a nonexistent file, reading from a file or running into issues when writing data. Alternative actions can be implemented by enclosing file operations in a try block and addressing particular exceptions such as 'FileNotFoundError', 'PermissionError' or 'IOError' in the except block as shown in Fig 3.14.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
1  try:
2      with open("abc.txt", "r") as file:
3          for char in file.read():
4              print(char, end="")
5      print("\nFile read successfully")
6  except FileNotFoundError:
7      print("Error: The file does not exist.")
8  except IOError:
9      print("Error: An issue occurred while reading the file.")
```

Fig.3.14: try-except block

Activity 6

Write a program to open the file 'abc.txt' in read-mode. Create another file 'xyz.txt' and write the contents of abc.txt by reading character by character.

3.2.8 GUI in Python

Graphical User Iinterface (GUI) improves the usability and accessibility of applications as compared to Command-Line Interface (CLI) systems which require users to write commands. In GUI through components like menus, text fields, buttons etc., users may interact with the program more smoothly. Python provides a library for developing GUIs and games namely 'Tkinter to handle common tasks like user input i.e. the steps to be taken when a button is clicked. Tkinter is a built-in Python library used to create GUIs. It allows programmers to create interactive application components, like desktop Applications including word processors, media players and basic utilities where all their interfaces can be made with Tkinter. Tkinter also can be used for creating 2D games with simple graphics. Game elements and windows can be created and user inputs like keyboard and mouse clicks can be controlled using functions provided by Tkinter. Options provided by Tkinter, include:

- The basic components of the interface are called widgets which include checkboxes, buttons, labels and text boxes. We can organize and modify widgets to make the layout as desired.
- Tkinter also is used for event handling like mouse clicks, key presses and window resizing. We can use code to create responses to these events for interactive and dynamic applications.

The arrangement of GUI components and the underlying logic are generally kept separate in the code.

Simple Number Adder with GUI

Before moving on with the code, it is quite helpful to understand the steps involved in creating a GUI using Tkinter library. Let us create a simple Adder application which takes in two numbers as input and when the 'Add' button is pressed, the result gets displayed. Major steps are:

1. First of all, import the tkinter library in Python code and create the main application window which will hold all

```
import tkinter as tk
```


Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

components and is created using tkinter.Tk().

2. Next we define a function to add numbers on the screen.

```
1   import tkinter as tk
2
3   def add_numbers():
4       try:
5           num1 = float(entry1.get())
6           num2 = float(entry2.get())
7           result_label.config(text=f"Sum: {int(num1 + num2)}")
8       except ValueError:
9           result_label.config(text="Invalid input. Please enter numbers."
```

We take in the user input into the first and second boxes using get() function. Next we add the numbers and update the result label to display the sum. If the user types something that's not a number like letters, etc. an error message is displayed.

3. Thereafter we defined another function to clear fields.

```
11  def clear_fields():
12      entry1.delete(0, tk.END)
13      entry2.delete(0, tk.END)
14      result_label.config(text="Result will be displayed here.")
```

To clear the contents of the first text entry box, we use the delete command, starting at the 'position 0' and going all the way to the end. Next, we do the same thing for the second text entry box, clearing its text as well. Finally, we reset the result label to its default message by configuring its text to "...".

4. Create main window along with the title of the program should be the next step using:

```
16  root = tk.Tk()
17  root.title("Simple Adder")
```

5. As a next step we setup input fields:

```
19  entry1 = tk.Entry(root, width=10)
20  entry1.grid(row=0, column=0, padx=5, pady=5)
21
22  entry2 = tk.Entry(root, width=10)
23  entry2.grid(row=0, column=1, padx=5, pady=5)
```

We create a little text box for the user to type in. The 'tk.Entry()' creates the box itself and we have set it to be 10 characters wide. Then using 'grid()' we put it in the top left corner of our window (row 0 and column 0 to be specific). We also added a little bit of space around it i.e. 5 pixels on all sides using padx and pady. Similarly for the second text box next to the first one, in the same row (row 0) but one column over (column 1).

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

6.Subsequently buttons are added in the screen.

```
25    add_button = tk.Button(root, text="Add", command=add_numbers)
26    add_button.grid(row=1, column=0, padx=5, pady=5)
27
28    clear_button = tk.Button(root, text="Clear", command=clear_fields)
29    clear_button.grid(row=1, column=1, padx=5, pady=5)
```

We create an "Add" button which when clicked will call "add_numbers()" function. This button is placed in the second row and the first column of our window with a little bit of space around it as in the last step. We also have a "Clear" button that when clicked will call "clear_fields()" function.

7.To display result, a label needs to be added. So, a label is created to show the results. Initially, it says "Result will be displayed here." This label is positioned on the grid in row 2, stretching across both columns using the 'columnspan' attribute. We also add a little bit of padding around the label to make it look nicer.

```
31    result_label = tk.Label(root, text="Result will be displayed here.")
32    result_label.grid(row=2, column=0, columnspan=2, padx=5, pady=5)
```

8.Lastly we run the GUI in a continuous manner. Without it, the window will just flash and disappear. This loop ensures that the windows stays open and interactive.

```
34    root.mainloop()
```

The complete code will look like as shown in Fig 3.15.

```
1   import tkinter as tk
2
3   def add_numbers():
4       try:
5           num1 = float(entry1.get())
6           num2 = float(entry2.get())
7           result_label.config(text=f"Sum: {int(num1 + num2)}")
8       except ValueError:
9           result_label.config(text="Invalid input. Please enter numbers.")
10
11  def clear_fields():
12      entry1.delete(0, tk.END)
13      entry2.delete(0, tk.END)
14      result_label.config(text="Result will be displayed here.")
15
16  root = tk.Tk()
17  root.title("Simple Adder")
18
19  entry1 = tk.Entry(root, width=10)
20  entry1.grid(row=0, column=1, padx=5, pady=5)
```


Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
21
22    entry2 = tk.Entry(root, width=10)
23    entry2.grid(row=0, column=1, padx=5, pady=5)
24
25    add_button = tk.Button(root, text="Add", command=add_numbers)
26    add_button.grid(row=1, column=0, padx=5, pady=5)
27
28    clear_button = tk.Button(root, text="Clear", command=clear_fields)
29    clear_button.grid(row=1, column=1, padx=5, pady=5)
30
31    result_label = tk.Label(root, text="Result will be displayed here.")
32    result_label.grid(row=2, column=0, columnspan=2, padx=5, pady=5)
33
34    root.mainloop()
```

Output

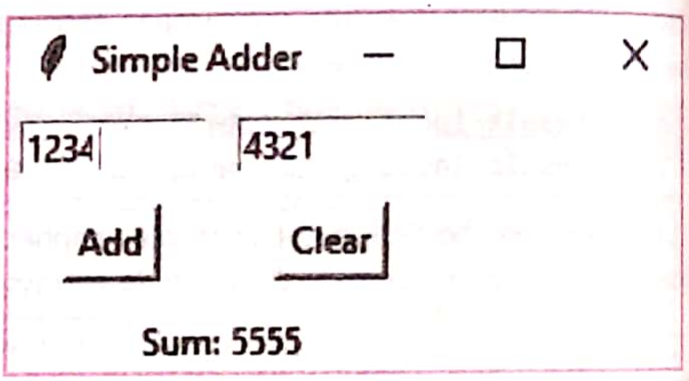


Fig.3.15 : tkinter

Developing Game using tkinter

In terms of GUI application, even games can be developed using tkinter. Let us create a simple game of correctly identification of color such that the text color needs to be identfied and written with correct spellings. The twist however is that the text show name of a different color to confuse the player. Only right color input with correct spellings will be counted.

Steps to create the color game:

1. Create a new Python file and import necessary libraries:
    1. import tkinter as tk
    2. import random
2. Set a list of colors:

colors = ["red", "blue", "green", "yellow", "pink", "black", "orange", "purple", "brown"]

3. Sketch game logic

- Initialize variables like score = 0.
- Outline 'new_round()' function:
    - Select a random color from the colors list for 'color_to_guess'.
    - Select another random color for the text's display color 'random_color'.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- Align the label with the 'color_to_guess' text and 'random_color'.
- Clear player's input from the entry field, if any.

- For 'check_answer()' function:
  - Acquire the user_input from the entry field and check if it matches the random_color.
  - If correct, increment the score and update the score_label.
  - If incorrect, display a message showing incorrect answer.
  - Call 'new_round()' to start the next round.

4. GUI

- Create the main window:

1. window = tk.Tk()
2. window.title("Color Game")

- GUI elements that we need to place:

a) Add label 'Display the word to be guessed'.

```
27    label = tk.Label(window, font=("Arial", 24))
28    label.pack(pady=20)
```

b) Input field for entering answer.

```
30    entry = tk.Entry(window, font=("Arial", 18))
31    entry.pack()
```

c) Add a button to check player's answer when pressed.

```
33    check_button = tk.Button(window, text="Check", command=check_answer)
34    check_button.pack(pady=10)
```

d) Displays the player's current score.

```
36    score_label = tk.Label(window, text="Score: 0")
37    score_label.pack()
```

5. Starting the game

- Initially 'new_round()' function will be called to start the first game round.
- Call 'window.mainloop()' to run the game.

The complete code is shown in Fig 3.16.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
1  import tkinter as tk
2  import random
3
4  window = tk.Tk()
5  window.title("Color Game")
6  score = 0
7  colors = ["red", "blue", "green", "yellow", "pink", "black",
   "orange", "purple", "brown"]
8
9  def new_round():
10     global score
11     global color_to_guess, random_color
12     color_to_guess = random.choice(colors)
13     random_color = random.choice(colors)
14     label.config(text=color_to_guess, fg=random_color)
15     entry.delete(0, tk.END)
16
17 def check_answer():
18     global score
19     user_input = entry.get().lower()
20     if user_input == random_color:
21         score += 1
22         score_label.config(text="Score: " + str(score))
23     else:
24         score_label.config(text="Score: " + str(score) + "
           (Incorrect)")
25     new_round()
26
27 label = tk.Label(window, font=("Arial", 24))
28 label.pack(pady=20)
29
30 entry = tk.Entry(window, font=("Arial", 18))
31 entry.pack()
32
33 check_button = tk.Button(window, text="Check",
   command=check_answer)
34 check_button.pack(pady=10)
```

① ② ③ ④ ⑤

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
35
36    score_label = tk.Label(window, text="Score: 0")
37    score_label.pack()
38
39    new_round()
40    window.mainloop()
```

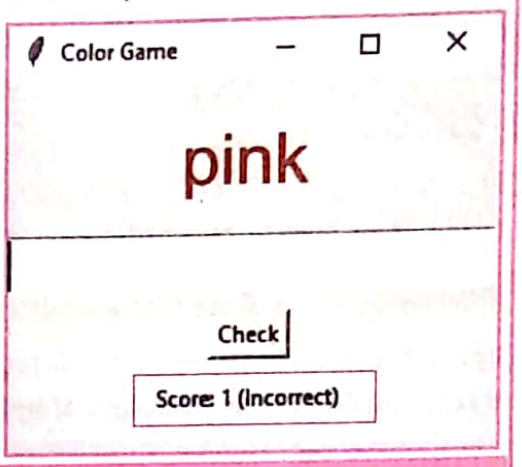


Fig.3.16 : tikinter

3.2.9 Databases

A set of data organized in a systematic manner, usually in a computer system, is called a database. A database stores, retrieves and handles large number of records for multiple users. Databases allow users to locate specific information quickly and accurately. The central source of data for every software program is a database. For instance, when a user logs in, shops online or makes a transaction, a database system saves the relevant data so that it may be retrieved later, i.e. what time the user logged in, which items the user purchased or the transaction to which account took place, etc.

Databases remove redundancies and inconsistencies that are possible in manual record keeping techniques by organizing data and ensuring its reliability. Databases also provide concurrent access and changes which allows several users to work on the same data-set at the same time. They also include security measures to safeguard sensitive information from unwanted access. Databases provide effective information retrieval and help in data analysis.

Types of Databases

- Most databases are relational. They organize data into rows and columns of a table such that each table represents an individual entity (for example customers or items).
- NoSQL databases provide flexibility and can manage vast amounts of unstructured or semi-structured data. Unlike relational databases, they do not have a fixed table structure.

Key Concepts involved in databases are:

- **Table:**

A table is the fundamental structure of a relational database and is made up of rows and columns. Each row denotes a record, while each column represents a field.

- **Record:**

A record is a single table item with information about a specific entity. For example a client record could include information like his or her name, address and telephone number.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- **Field:**

A field is a piece of data within a record that defines the information placed in a certain column. For example, a Name field would store text data, whereas an Age field would store numerical data.

- **Data Integrity**

Data integrity refers to the accuracy and consistency of data within a database. It ensures that the data is accurate, dependable and appropriate for usage.

Primary & Secondary Keys

Keys are essential in creating database to uniquely identify records and create links among tables. The most common types of keys used in normalization are:

- **Primary Key:**

A primary key is a minimal set-of-attributes that is exclusively able to identify a record. It can be a single attribute or a combination of attributes.

- **Candidate Key:**

A candidate key is a single or a set-of-attributes that allow identifying of a record uniquely. Though there may be several candidate keys, only one is chosen to be the primary key. The rest are called secondary keys.

- **Foreign Key:**

A foreign key is a field in one table that creates a relationship with another table by referring to its primary key.

- **Composite Key:**

A composite key is a combination of two or more attributes that distinctively identify a row in a database table.

Data Normalization

A key component of relational database management is data normalization, which focuses on minimizing duplicate data. To ensure effective and consistent data storage and speed-up database access, the contents are divided into smaller interlinked tables. Data redundancy or duplicate storage of the same information is a common problem in databases which causes problems in maintaining consistency and increase chances of inaccuracies during updating of data. By dividing the data into more manageable and inter-linked tables, normalization solves these problems. Every table represents a separate entity with keys defining the relationships among the entities.

- Without normalization databases frequently show inconsistencies and abnormalities during updates or deletions. For example, a database containing customer information with several addresses would require changes in many locations for every address change, increasing the chances of mistakes. Normalized data addresses this by creating various tables for customers and addresses, which are connected by a key like customer-ID.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Normalization steps for a database are:

- **First Normal Form (1NF):**

In 1NF tables have atomic values along with repeating groupings /multi-valued cells are removed.

- **Second Normal Form (2NF):**

2NF ensures that every non-key attribute is dependent on the whole Primary key.

- **Third Normal Form (3NF):**

In 3NF transitive dependencies, in which one non-key attribute relies upon another non-key attribute, are eliminated. For example, while developing a small database for a laptop shop the initial information may look like:

For a laptop shop if we want to create a database, the steps will be as follows:

Step 1:

Classify entities along with the attributes, which are:

- Customer: C_ID, C_Name, C_Address, C_Phone
- Laptop: L_ID, L_Name, L_Price, L_Brand
- Order: OrderID, OrderDate, C_ID, L_ID, Quantity

Step 2:

Let us initially create a single table to represent all the data:

OrderID	OrderDate	C_ID	C_Name	C_Address	C_Phone	L_ID	L_Name	L_Price	L_Brand	Quantity
1	20-11-2024	C1	Qamar	House:123	123-4567	L1	Dell XPS 13	100000	Dell	1
2	21-11-2024	C2	Mahtab	House: 456	987-6543	L2	MacBook Air	90000	Apple	2
3	22-11-2024	C3	Qamar	House:123	123-4567	L3	HP Spectre x360	120000	HP	1
4	23-11-2024	C4	Mahtab	House: 456	987-6543	L4	Dell XPS 13	100000	Dell	1
5	24-11-2024	C5	Chaand	House: 789	543-2109	L5	MacBook Air	90000	Apple	2

Step 3:

To normalize to 1NF we need to find repeating groups, which are:

Customer information that is repeated for each order.

Laptop information that is repeated for each order.

So, let us create separate tables as follows:

- Customer (C_ID, C_Name, C_Address, C_Phone)
- Laptop (L_ID, L_Name, L_Price, L_Brand)
- Order (OrderID, OrderDate, C_ID, L_ID, Quantity)

Hence, 1NF Tables are:

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Customer

C_ID	C_Name	C_Address	C_Phone
C1	Qamar	House:123	123-4567
C2	Mahtab	House: 456	987-6543
C3	Chaand	House: 789	543-2109

Laptop

L_ID	C_Name	L_Price	L_Brand
L1	Dell XPS 13	100000	Dell
L2	MacBook Air	90000	Apple
L3	HP Spectre x360	120000	HP

Order

OrderID	OrderDate	C_ID	L_ID	Quantity
1	20-11-2024	C1	L1	1
2	21-11-2024	C2	L2	2
3	22-11-2024	C1	L3	1
4	23-11-2024	C2	L1	1
5	24-11-2024	C3	L2	2

Step 4:

To normalize to 2NF we need to detect partial dependencies, which come out to be:

- OrderID depends on OrderDate, but not on C_ID or L_ID.

As a result, we need to create a new table:

- OrderDetails (OrderID, OrderDate)

As a result, 2NF Tables are:

OrderDetails

OrderID	OrderDate
1	20-11-2024
2	21-11-2024
3	22-11-2024
4	23-11-2024
5	24-11-2024

Order

OrderID	C_ID	L_ID	Quantity
1	C1	L1	1
2	C2	L2	2
3	C1	L3	1
4	C2	L1	1
5	C3	L2	2

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Step 5:

Lastly, to normalize to 3NF we ought to identify transitive dependencies, i.e.:

- L_ID in the Order table depends on L_Name, L_Price, and L_Brand, which are attributes of the Laptop table.

To remove transitive dependencies:

- The Order table now only references the L_ID.

Thus, the 3NF Tables are:

OrderDetails

OrderID	OrderDate
1	20-11-2024
2	21-11-2024
3	22-11-2024
4	23-11-2024
5	24-11-2024

Customer

OrderID	C_ID	L_ID	Quantity
1	C1	L1	1
2	C2	L2	2
3	C1	L3	1
4	C2	L1	1
5	C3	L2	2

Order

C_ID	C_Name	C_Address	C_Phone
C1	Qamar	House:123	123-4567
C2	Mahtab	House:456	987-6543
C3	Chaand	House:789	543-2109

Laptop

L_ID	L_Name	L_Price	L_Brand
L1	Dell XPS 13	100000	Dell
L2	MacBook Air	90000	Apple
L3	HP Spectre x360	120000	HP

Database Tools

Various database tools are available for data management, with Microsoft Access and SQLite being two popular options. MS Access, part of the MS Office suite, is simple to install, easy to use, and offers a user-friendly drag-and-drop interface. It is commonly used by individuals and small organizations for tasks like inventory, project management, and event planning. MS Access also integrates well with other MS Office programs like Excel and Project.SQLite is a lightweight database system that stores data in a single file. It is ideal for applications needing local storage and is widely used in mobile apps and scenarios that do not require a large database server. SQLite is simple, efficient, and easy to use.

Advantages of SQLite:

- SQLite occupies small space making it suitable for applications with limited resources.
- It is easily integrated into applications, hence becomes easier for use and deployment.
- SQLite has a serverless architecture which eliminates unnecessary dependencies.
- The complete database is saved in a single file.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.2.10 Create a Database

1. Installation of SQLite

To download and install SQLite on your Windows system, first go to the official SQLite website (https://sqlitebrowser.org/dl/). Look for a "Download" button and select the right download for your system. SQLite normally comes with already compiled files. Select the version that matches your Windows operating system and click the download link. The browser will prompt you to save the file. Choose a suitable location on your computer, such as the 'Downloads' folder. After the download is finished, you will notice that the file is a compressed (e.g., a ZIP) file. To access the SQLite files, right-click the downloaded file and select "Extract All". This will create a new folder containing the SQLite files which are ready to be used.

To check whether the SQLite was successfully installed, launch the 'Command Prompt' by typing "cmd" into the Windows search bar and click it. After opening the Command Prompt, use the 'cd' command to navigate to the folder where you extracted the SQLite files (for example, 'cd C:\Downloads\sqlite\'). Next type'sqlite3' and click Enter. If the installation was successful, SQLite's command prompt will appear showing that it is ready for use.

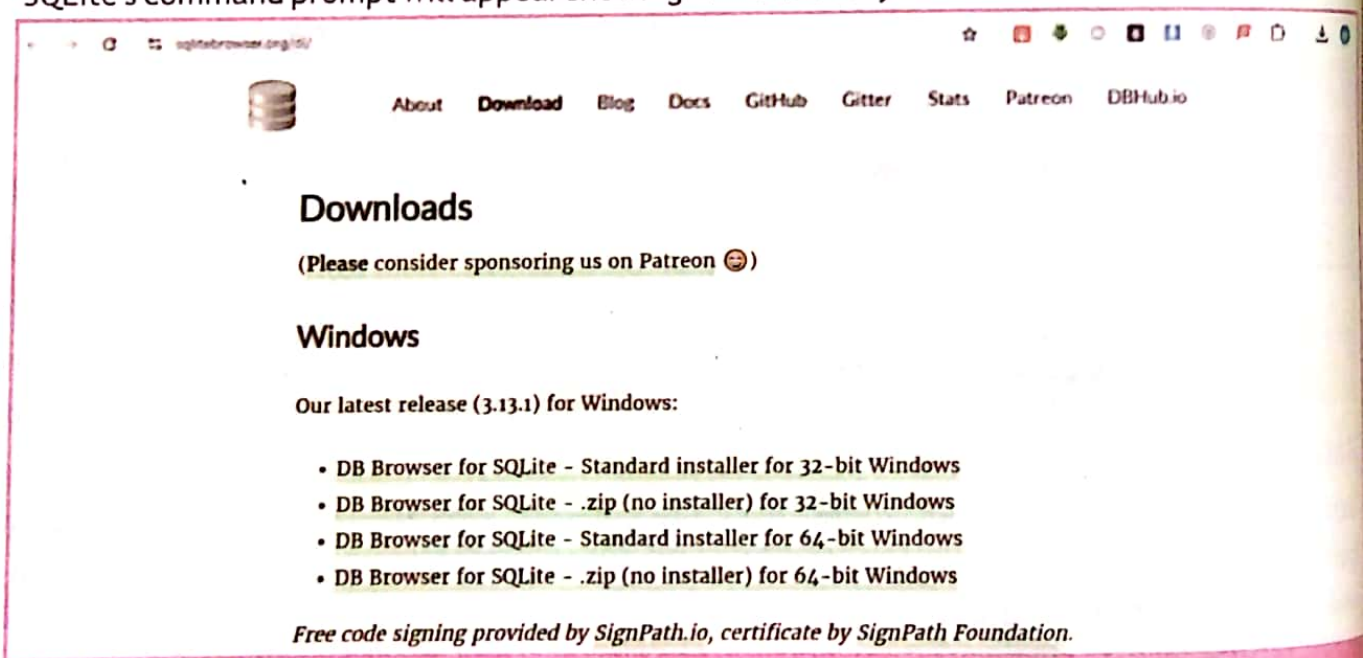

About
Download
Blog
Docs
GitHub
Gitter
Stats
Patreon
DBHub.io
Downloads
(Please consider sponsoring us on Patreon)
Windows
Our latest release (3.13.1) for Windows:
• DB Browser for SQLite - Standard installer for 32-bit Windows
• DB Browser for SQLite - .zip (no installer) for 32-bit Windows
• DB Browser for SQLite - Standard installer for 64-bit Windows
• DB Browser for SQLite - .zip (no installer) for 64-bit Windows
Free code signing provided by SignPath.io, certificate by SignPath Foundation.

Fig.3.17 : Installation

2. Database Creation

To create a new SQLite database, launch the SQLite Command-Line Interface (CLI). To access the interactive SQLite shell:

- Open your terminal or command prompt and type'sqlite3' and press Enter.
- Once inside the SQLite shell, use the command '.open NBF.db' to create a new database named 'NBF'.

This command creates a new file, 'NBF.db' in the current directory to store your database. And we have successfully created a new SQLite database.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

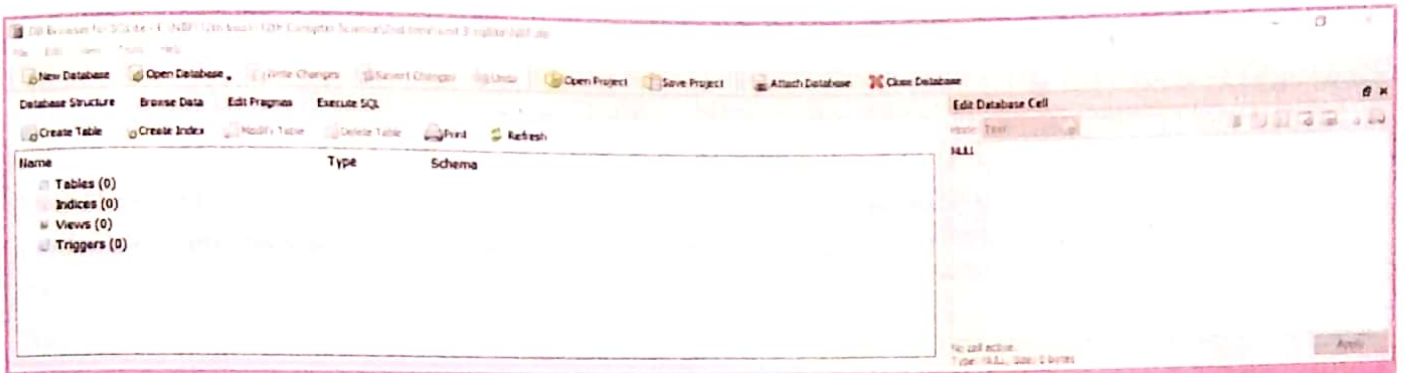


Fig.3.18 : Database Creation

3. Creating a Table

As a next step, use the 'CREATE TABLE' command to construct a table in the database. This command takes in the table's name, column names and data types. For example, we can create a table called 'Laptop' with four columns: 'L_ID' (an integer primary key), 'L_Name' (text), 'L_Price' (integer) and 'L_Brand' (text). So far only the schema is created and it does not contain any record.

```
1  CREATE TABLE Laptop (
2      L_ID TEXT PRIMARY KEY,
3      L_Name TEXT NOT NULL,
4      L_Price INTEGER NOT NULL,
5      L_Brand TEXT NOT NULL
6  );
```

Output

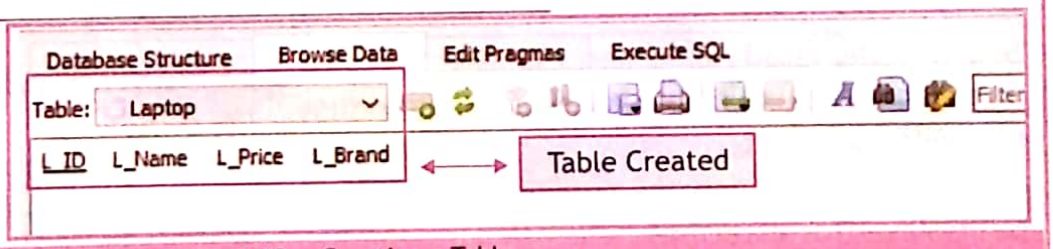


Fig.3.19 : Creating a Table

4. Adding Record(s)

Once the table is created we can use the 'INSERT INTO' command to enter a single row in the table. This command requires the table name, column names and values for the record we wish to add in the table. Be careful to use the values according to the schema, sequence matters.

```
1  INSERT INTO Laptop (L_ID, L_Name, L_Price, L_Brand)
2  VALUES ('L1', 'Dell XPS 13', 100000, 'Dell');
```

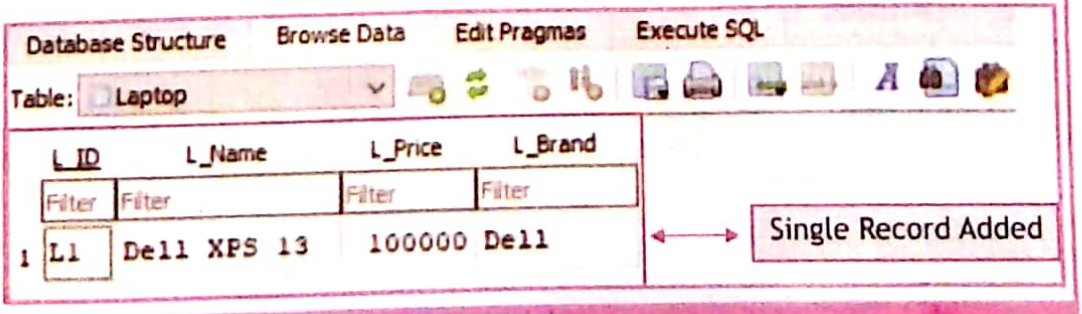


Fig.3.20 :Adding single record

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Rather than adding single record, we can add multiple records as well using a single 'Insert Into' command as shown in Fig 3.21.

```
1   INSERT INTO Laptop (L_ID, L_Name, L_Price, L_Brand)
2   VALUES
3       ('L2', 'MacBook Air', 90000, 'Apple'),
4       ('L3', 'HP Spectre x360', 120000, 'HP');
5
```

Output

Database Structure | Browse Data | Edit Pragmas

Table: Laptop

	L_ID	L_Name	L_Price	L_Brand
	Filter	Filter	Filter	Filter
1	L1	Dell XPS 13	100000	Dell
2	L2	MacBook Air	90000	Apple
3	L3	HP Spectre x360	120000	HP

Multiple Record Added

Fig.3.21 :Adding multiple records

In this fashion, we can create additional tables and populate the data.

5. Querying Records from Table using Select

'Select' is the mostly used query to select records from the table. An '*' means all records in that particular table. Alternatively, we can mention the column names and only the specified columns will be displayed for all records.

Though select query fetches all records from the table but we can apply condition using where clause to get matching records. The matching can be a single or multiple records satisfying the condition.

```
1   SELECT * FROM Customer;
2
```

Output

	C_ID	C_Name	C_Address	C_Phone
1	C1	Qamar	New Address	123-4567
2	C2	Mahtab	House: 456	987-6543
3	C3	Chaand	House: 789	543-2109

Select All Records

Fig.3.22 : Querying Records from Table using Select

```
1   SELECT * FROM Customer
2   WHERE C_ID = 'C2';
3
```

	C_ID	C_Name	C_Address	C_Phone
1	C2	Mahtab	House: 456	987-6543

Record with ID 'C2'

Output

Fig.3.23 : Select query fetches all records from the table

Activity 7

Populate data for tables 'Customer', 'Order' and 'OrdreDetails'.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

6. Update Record

For updating a record or a particular column, 'update' command is used. A good practice is to use select statement with where clause so that the desired record is retrieved. Thereafter for the same condition use the update command.

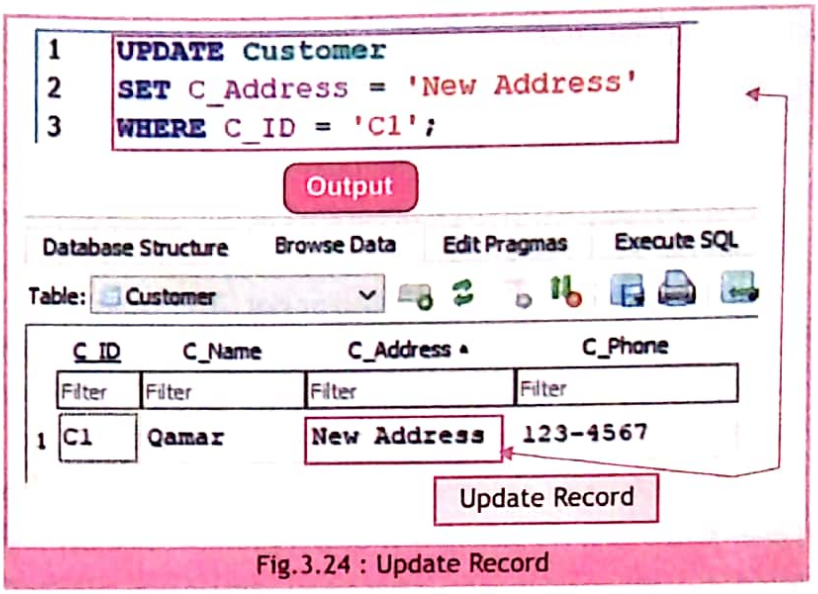


Fig.3.24 : Update Record

7. Use of Order By Clause

There are scenarios where records should be displayed in a sorted order. For that reason 'order by' clause is used in the query and the result will be displayed in ascending order, by default. An additional 'DESC' keywords is used with 'order by' clause to display the results in descending order.

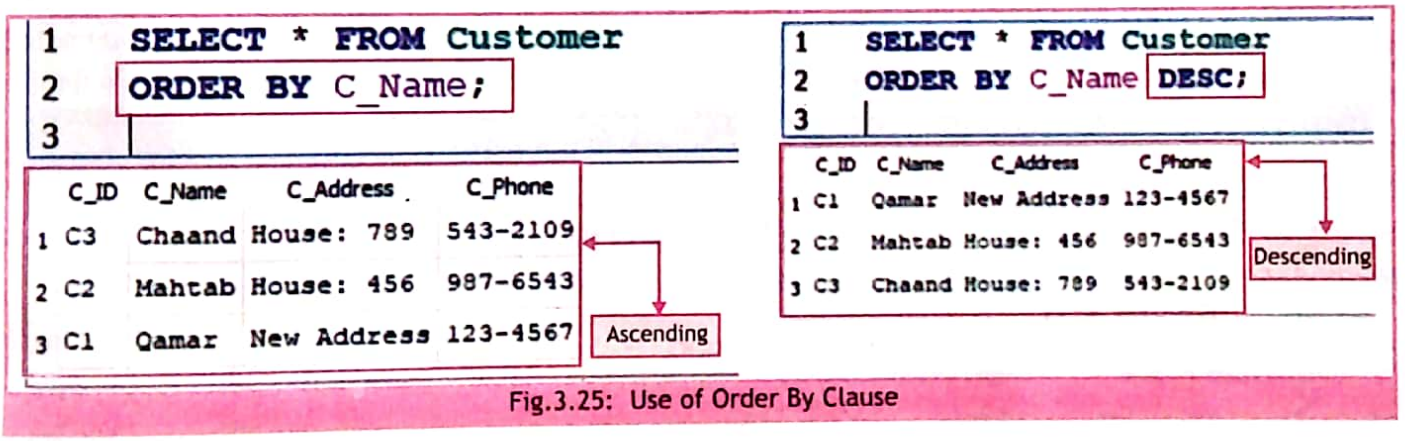

1 SELECT * FROM Customer
2 ORDER BY C_Name;
3
C_ID C_Name C_Address C_Phone
1 C3 Chaand House: 789 543-2109
2 C2 Mahtab House: 456 987-6543
3 C1 Qamar New Address 123-4567
Ascending
1 SELECT * FROM Customer
2 ORDER BY C_Name DESC;
3
C_ID C_Name C_Address C_Phone
1 C1 Qamar New Address 123-4567
2 C2 Mahtab House: 456 987-6543
3 C3 Chaand House: 789 543-2109
Descending

Fig.3.25: Use of Order By Clause

8. Deletion of Record(s)

You may delete a single or multiple entries from a table using 'Delete' command. However the table structure will remain in the database. It works just like select statement with an '*' all records will be deleted while using 'where' clause only the selected record(s) will be deleted.

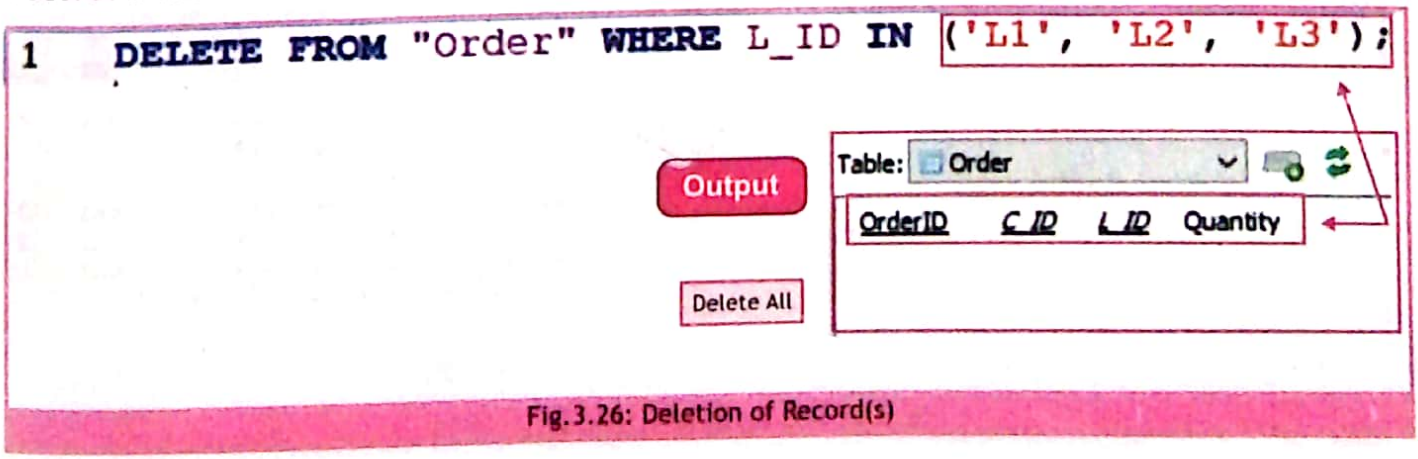


Fig.3.26: Deletion of Record(s)

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.3 Techniques for Testing & Debugging

3.3.1. Debugging

Debugging is the process to analyze the code for bugs and errors and help in locating them, such that when fixed error-free program can run smoothly and deliver desired results. The simplest of debugging involves adding multiple print statements in a program to get the idea whether the program is behaving as expected or not. This does not mean that we should add print statements on each line or every second line. The idea is that wherever some formulae, expression or a critical value update is in the code, using print statements it should be validated that particular variable, line or block of code is behaving as desired.

To make the point clearer, let us refer the matrix transpose example as shown in Fig. 3.27. Here on line 8, we have intentionally placed a bug, by switching the positions of the matrix indices. Once we run it the output shows that matrix and transpose matrix are same. So, within the nested loop, we have placed multiple print statements which we not only label the print statement for identification purposes, but also print the values of different variables and data structures. For example, 'Print-1' on line 6 ensures that the 'new_row' gets always initialized whenever the outer loop index updates. Similarly, 'Print-3' on line 10 prints the values which will be appended in the transpose matrix after completion of the inner loop. On line 9, 'Print-2' is placed inside the nested for-loop to print the values of matrix indices and points out that the values of matrix are as it is assigned to transpose and needs to be changed.

```
1  matrix = [[1, 2, 3],[4, 5, 6],[7, 8, 9]]
2  transpose = []
3
4  for col in range(len(matrix[0])):
5      new_row = []
6      print('Print1-Outer Loop: ',new_row)
7      for row in range(len(matrix)):
8          new_row.append(matrix[col][row])
9          print('Print2-Inner Loop: row: ',row,'- col: ', col)
10     print('Print3-Outer Loop: ',new_row)
11     transpose.append(new_row)
12 for row in transpose:
13     print(row)
```

Output

```
Print1-Outer Loop:  []
Print2-Inner Loop: row:  0 - col:  0
Print2-Inner Loop: row:  1 - col:  0
Print2-Inner Loop: row:  2 - col:  0
Print3-Outer Loop:  [1, 2, 3]
Print1-Outer Loop:  []
Print2-Inner Loop: row:  0 - col:  1
Print2-Inner Loop: row:  1 - col:  1
Print2-Inner Loop: row:  2 - col:  1
Print3-Outer Loop:  [4, 5, 6]
Print1-Outer Loop:  []
Print2-Inner Loop: row:  0 - col:  2
Print2-Inner Loop: row:  1 - col:  2
Print2-Inner Loop: row:  2 - col:  2
Print3-Outer Loop:  [7, 8, 9]
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
```

Fig.3.27: Debugging with Print Statements

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

This code performs the transpose of a 3x3 matrix (swapping rows with columns). Here's a simple explanation:

- Line 2-3: Initializes a matrix and an empty list transpose to store the result of the transposed matrix.
- Line 5-8: The outer loop goes through each column in the original matrix. Inside this loop, a new row (new_row) is created.
- Line 9-12: The inner loop iterates through each row of the matrix, appending the corresponding column value to new_row.
- Line 14-15: Once the inner loop finishes, the new_row (a transposed row) is added to transpose.
- Line 16-17: Finally, the code prints the transposed matrix by printing each row in transpose.

Advanced debugging involves more than basic print commands by using techniques like breakpoints and watchpoints to identify and fix bugs and errors. Breakpoints are placed on the line of code while watchpoints are set on variables and expressions or formulae. Multiple breakpoints and watchpoints can be placed in a program. These methods are especially useful for handling complex issues.

The same code of Fig.3.27 can be placed in a debug mode in an IDE like Visual Studio.Net or an online IDE like REPLIT which provide a better GUI interface like selecting the line and setting the breakpoint on the said line. When run in debug mode step by step, the debugger not only stops at the breakpoint but also provides the values of the different variables and data structures. This helps in analyzing the data and overall behavior of the program. Fig. 3.28 shows the code while the debugger is on, reflecting the different variable values and the corresponding breakpoint set at line 6.

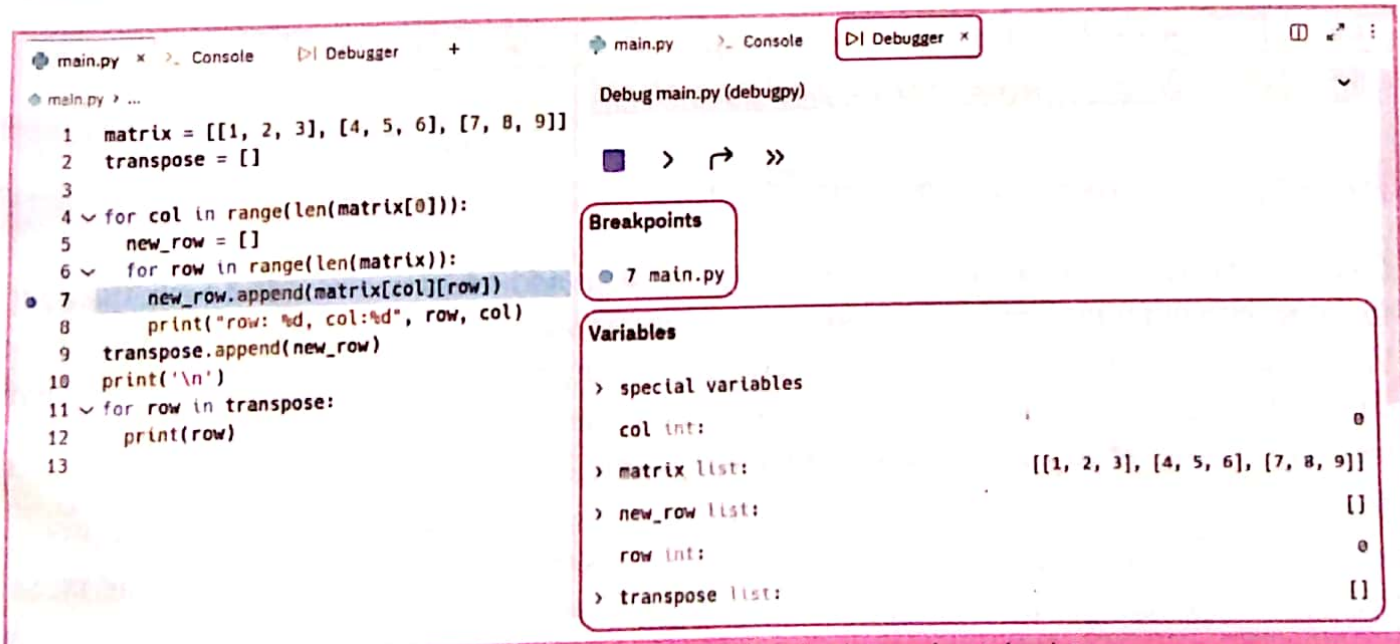


Fig.3.28: Debugging the code with breakpoints and watchpoints

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3.3.2 Python Unit test

Python's built-in unittest framework is a tool that helps you check if your code works in the desired way. Rather than manually testing your code by running it, unittest lets you write small tests that check specific parts of your program. You may check it for some values to check whether it is returning the desired and correct result. But as the lines of code increase the code may become compilcated to test it manually. unittest is quite useful in such scenarios to write a set of tests to ensure the function is executing correctly for various inputs.

So, for a written function that performs some operation, you would write another function (a test function) to check if the first function gives the correct result. Use assertions to compare the actual output with the expected output. For example, you have a function that adds two numbers, like:

```
1   def add(a, b):                  # Function to be tested
2       return a + b
3
4   def test_add():                 # Test function
5       result = add(2, 3)          # Test case 1: Check if 2 + 3 equals 5
6       assert result == 5, f"Test failed: Expected 5 but got {result}"
7
8       result = add(0, 0)          # Test case 2: Check if 0 + 0 equals 0
9       assert result == 0, f"Test failed: Expected 0 but got {result}"
10
11      result = add(-1, 1)         # Test case 3: Check if -1 + 1 equals 0
12      assert result == 0, f"Test failed: Expected 0 but got {result}"
13
14      print("All tests passed!")
15
16  test_add()                      # Run the test function
```

Fig.3.29: Function to add 2 numbers with tests

On line 1 the add() function adds two numbers. The test_add() function on line 4 tests the add() function with different inputs, like as input parameters two positive numbers are provided on line 5, two null values on line 8 and a positive and negative number on line 11. The assert statement checks if the output of add() matches the expected value. If not, it raises an error with a custom message. If all assertions pass, it prints "All tests passed!"

To have a better understanding you may check if for different values and results. As a next step, we assume that expected result was 2 on line 12 and we oversaw the negative value on line 11. So, definitely we would get the 'Test failed: ...' message. To investigate it we can debug the code by applying breakpoints in the code:

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```python
def add(a, b):                  # Function to be tested
    return a + b

def test_add():                 # Test function
    result = add(2, 3)          # Test case 1: Check if 2 + 3 equals 5
    assert result == 5, f"Test failed: Expected 5 but got {result}"

    result = add(0, 0)          # Test case 2: Check if 0 + 0 equals 0
    assert result == 0, f"Test failed: Expected 0 but got {result}"

    result = add(-1, 1)         # Test case 3: Check if -1 + 1 equals 0
    assert result == 0, f"Test failed: Expected 0 but got {result}"

    print("All tests passed!")

test_add()                      # Run the test function
```

Fig.3.30: Adding Breakpoints

Let's execute the code and debug it. We deployed 3 breakpoints on line 6, 9 and 12 to track the values to be checked. As watchpoint we added 'result' variable. As the debugging starts and it stops on first breakpoint, the result is as expected.

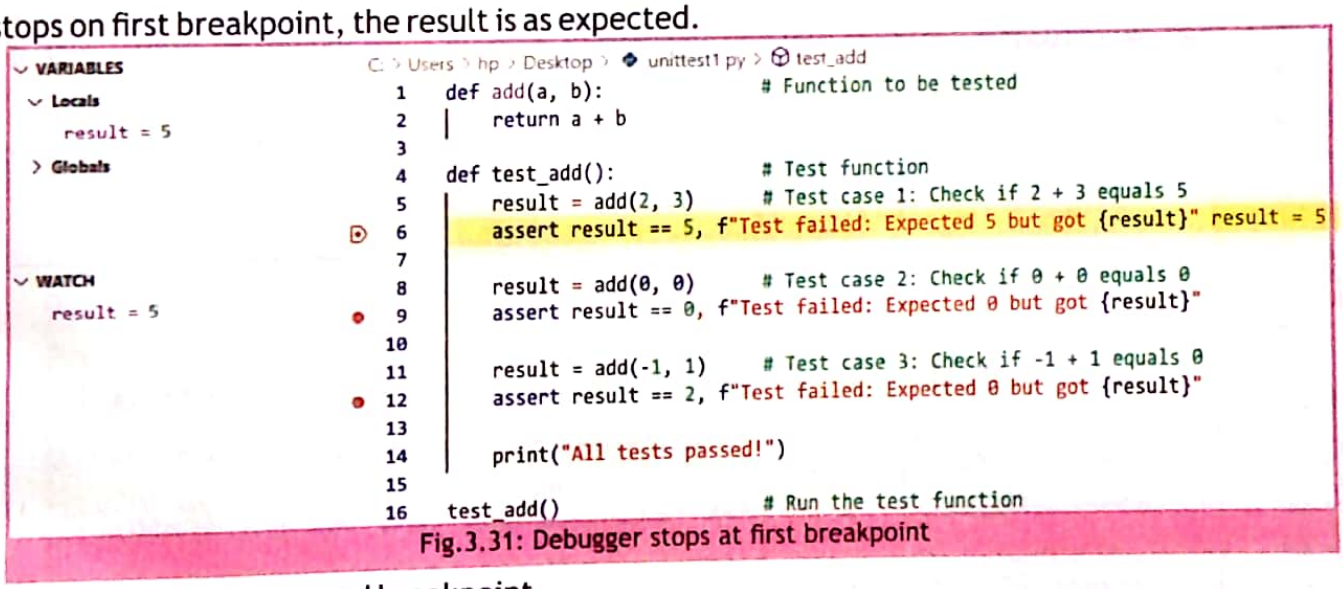

VARIABLES
Locals
result = 5
Globals
WATCH
result = 5
C: > Users > hp > Desktop > unittest1.py > test_add
assert result == 5, f"Test failed: Expected 5 but got {result}" result = 5

Fig.3.31: Debugger stops at first breakpoint

Same goes with the second breakpoint.

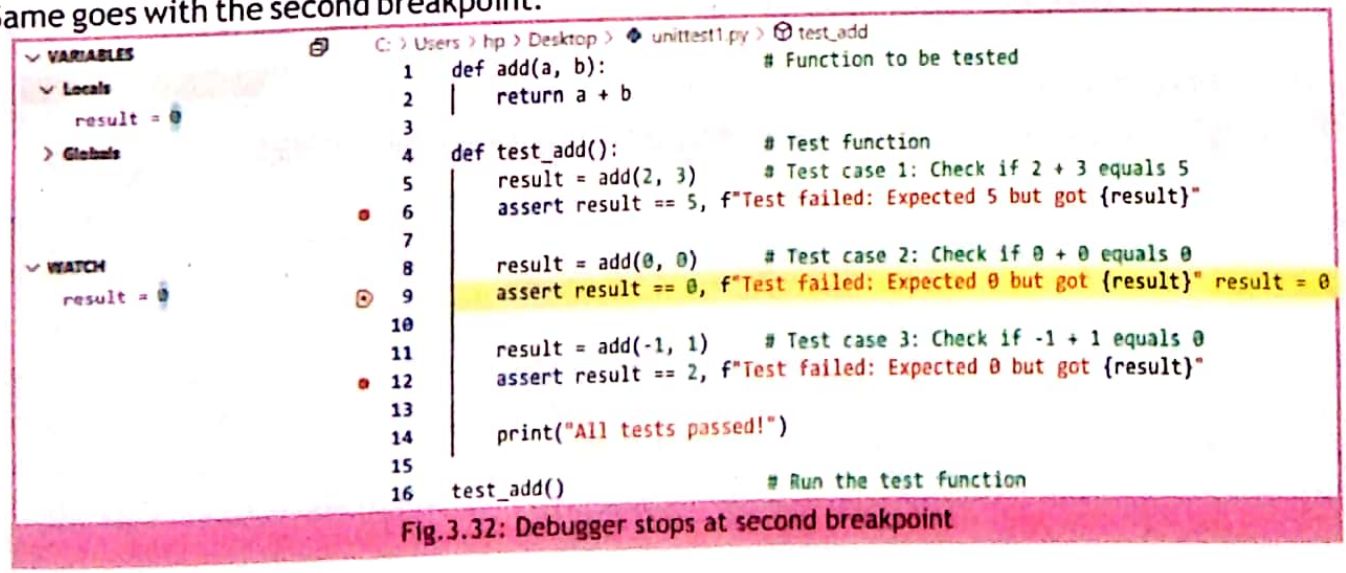

VARIABLES
Locals
result = 0
Globals
WATCH
result = 0
C: > Users > hp > Desktop > unittest1.py > test_add
assert result == 0, f"Test failed: Expected 0 but got {result}" result = 0

Fig.3.32: Debugger stops at second breakpoint

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

However, for the third breakpoint the result variable is flagged in the code and watchpoint shows the mismatched value as compared to the expected value to track the bug.

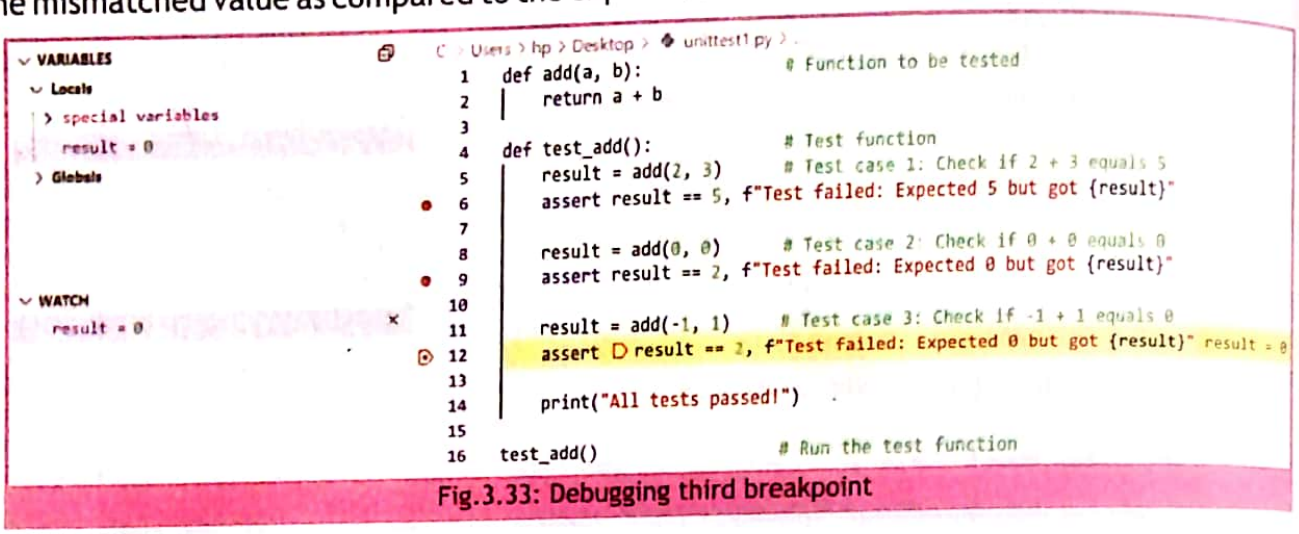

VARIABLES
Locals
special variables
result = 0
Globals
WATCH
result = 0
C > Users > hp > Desktop > unittest1.py > ...
1 def add(a, b): # Function to be tested
2 return a + b
3
4 def test_add(): # Test function
5 result = add(2, 3) # Test case 1: Check if 2 + 3 equals 5
6 assert result == 5, f"Test failed: Expected 5 but got {result}"
7
8 result = add(0, 0) # Test case 2: Check if 0 + 0 equals 0
9 assert result == 2, f"Test failed: Expected 0 but got {result}"
10
11 result = add(-1, 1) # Test case 3: Check if -1 + 1 equals 0
12 assert result == 2, f"Test failed: Expected 0 but got {result}" result = 0
13
14 print("All tests passed!")
15
16 test_add() # Run the test function

Fig.3.33: Debugging third breakpoint

The same code is changed for multiplication and tests are adjusted accordingly. Will all tests be passed? If not, which one will fail and why? Try debugging the code.

```
1   def multiply(a, b):
2       return a * b
3
4   def test_multiply():
5       result = multiply(2, 3)
6       assert result == 6, f"Test failed: Expected 6 but got {result}"
7
8       result = multiply(4, 0)
9       assert result == 1, f"Test failed: Expected 1 but got {result}"
10
11      result = multiply(-1, -1)
12      assert result == 1, f"Test failed: Expected 1 but got {result}"
13
14      print("All tests passed!")
15
16  test_multiply()
```

Fig.3.34: Function to multiply 2 numbers with tests

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Programming Paradigm:** Refers to different styles and methodologies of programming used to solve problems, making code more structured and easier to manage.
- **Complexity Reduction:** Helps to break down complex problems into simpler parts resulting in easier to understand and solve.
- **Functional Programming:** programs are developed in the form of functions.
- **Object-Oriented Programming (OOP):** Revolves around the use of objects and classes to represent real-world entities.
- **List in Python:** A data structure that stores an ordered collection of items which can be accessed by their index. Lists are mutable i.e. their contents can be changed.
- **Dictionary in Python:** A key-value pair data structure used for mapping keys to values such that every key in a dictionary is unique and values can be accessed using their corresponding keys.
- **Files in Python:** File handling allows Python programs to read from and write to files and store data permanently. Common operations include opening, reading, writing and closing files.
- **Databases:** A structured collection of data used for efficient storage and retrieval. Tools like MySQL, SQLite, and PostgreSQL are commonly used for managing databases.
- **Data Normalization:** A database design technique aimed at reducing redundancy and dependency by organizing fields and tables of data.
- **Primary Key:** Is a unique identifier for records in a database table emphasizing each record can be distinctly retrieved or updated.
- **GUI in Python:** A graphical user interface that allows users to interact with applications visually, using elements like buttons, text fields and windows, etc. Python's pygame library is commonly used to create GUIs.
- **Nested Lists:** Lists within lists are nested lists that are used for storing and organizing complex data structures like matrices.
- **List as Value in Dictionary:** Python allows the use of lists as values in dictionaries such that multiple items can be stored under a single key.
- **Testing in Python:** Testing ensures the correctness of code.
- **Unit Test:** Unit testing focuses on testing small, isolated units of functionality, such as individual functions or methods.
- **Debugging:** The process of identifying and fixing bugs or errors in a program. Techniques include using print messages, where messages are printed to the console to help track the flow of a program, and more sophisticated tools like debuggers that allow stepping through the code line by line.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. Main purpose of the programming paradigm is _____ .
   a. Increase code length
   b. Complexity reduction
   c. Eliminate bugs
   d. Improve hardware performance
2. _____ programming paradigm emphasizes the use of functions without changing state.
   a. Procedural
   b. Functional
   c. Object-oriented
   d. Structural
3. Object-Oriented Programming primarily revolves around _____ of the following.
   a. Data structures
   b. Functions
   c. Objects
   d. Variables
4. In Python _____ data structure is used to store a collection of items.
   a. List
   b. Dictionary
   c. Tuple
   d. Set
5. A dictionary stores _____ in Python.
   a. Keys only
   b. Values only
   c. Key-value pairs
   d. Sequential data
6. An example of disk I/O _____ .
   a. Storing a variable in memory
   b. Reading a file from the hard disk
   c. Defining a function
   d. Looping through a list
7. To write content to a file in Python _____ method is used.
   a. file.write()
   b. file.insert()
   c. file.add()
   d. file.modify()
8. A primary key in a database is _____ .
   a. A key that has a default value
   b. A unique identifier for records
   c. A key for linking multiple databases
   d. A key used for temporary tables
9. A common database normalization goal is _____ .
   a. Reducing redundancy
   b. Maximizing data size
   c. Increasing the number of tables
   d. Storing duplicate records
10. _____ key is used to uniquely identify a record in a related table.
    a. Primary
    b. Secondary
    c. Foreign
    d. Unique
11. To create a simple GUI in Python, _____ library is used.
    a. Pygame
    b. OS
    c. Sys
    d. math
12. A nested list in Python is referred as _____ .
    a. A list containing another list
    b. A list of dictionaries
    c. A dictionary of lists
    d. A set containing multiple lists
13. The purpose of unit testing in Python is _____.
    a. Test individual units of code
    b. Test the entire program
    c. Debug the entire program
    d. Optimize the code for speed

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

14. The primary goal of debugging is _____.

a. Writing new code
b. Fixing errors in the code
c. Increasing program speed
d. Creating a graphical interface

15. Python function used to print debugging messages to the console _____.

a. print()
b. debug()
c. trace()
d. log()

16. Output of the following code will be:

```
1  my_dict = {"a": [1, 2], "b": [3, 4]}
2  print(my_dict["a"][1])
```

a. 1
b. 2
c. [1, 2]
d. [3, 4]

17. The code will print "Inner" _____ times.

```
1  for i in range(3):
2      for j in range(2):
3          print("Inner")
```

a. 2
b. 3
c. 5
d. 6

Give short answers to the following Short Response Questions (SRQs).

1. Elaborate programming language paradigm and why is it important for software development?
2. How do you interpret the concept of complexity reduction in programming? How does dividing code into modules help manage complexity?
3. Compare Functional Programming and Object-Oriented Programming (OOP) paradigms. Enlist main differences in their approach to problem-solving.
4. What is Disk I/O and why is it important for system performance?
5. Describe the difference modes in file handling in Python.
6. What is the benefit of using the 'with' keyword while working with files in Python?
7. How can you create a simple GUI application using Pygame? Enlist the steps involved.
8. Describe the use of a nested list in Python and how can it be used to represent a matrix?
9. What is data normalization and why is it used in database management?
10. What is debugging and why is it important in software development?
11. What seems to be the problem in:

```
1  Sample_dict = {"a": 1, "b": 2}
2  print(sample_dict["c"])
```

12. Final value of my_list after the following lines would be:

```
1  own_list = [1, 2, 3]:
2  own_list.append(4, 5)
```


Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Give long answers to the following Extended Response Questions (ERQs).

1. Write a Python function that calculates the factorial of a given number. Test the function with 5 different inputs.
2. Implement a Python class 'Rectangle' with methods to calculate the area and perimeter of a rectangle. Create instances of the class and print the perimeter of rectangle.
3. Write a Python program highlighting the use of functions to simplify a problem. For example, create a program that calculates the average of a list of numbers and by using separate functions to add all numbers, , marking repeated values and calculate the average of the list with and without repeating values.
4. Create a Python dictionary to store student names as keys and their grades as values. Write a program to add new students, update existing students' grades, and remove students from the dictionary.
5. Write a Python program to create a file named data.txt, write a string to it and then read the content of the file and display the output.
6. Create a Python program that appends a new line of text to an existing file. If the file does not exist, it should create a new file.
7. Write Python code using Pygame that creates a simple window with a button. When the button is clicked, it should display a message on the window.
8. Develop a Python program that calculates and prints the transpose of a 3x3 matrix represented using a nested list.
9. Write a Python function that counts the number of occurrences of each letter in a text file and prints the results.
10. Write a Python program to connect to a database and demonstrate how to insert, update and retrieve data from the database. Explain the importance of database connectivity in Python applications.
11. Explain unit testing and its importance in Python programming. How does it help in catching errors early during the development process? Include an example of a unit test for a simple Python function.
12. Write a Python code that selects a number between 1 and 10 and the player will try to guess the number in 5 tries. For every incorrect guess, the program provides a hint whether the last guess was on higher or lower side. If the number is successfully guessed then a Congratulatory message should be displayed along with in how many attempts the number was guessed correctly. Otherwise another message should be displayed along with the chosen number.

The outputs should look like:

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

```
Hello, What's your name?Abrar
okay! Abrar I am Guessing a number between 1 and 100:
50
Your guess is too low
75
Your guess is too high
60
Your guess is too high
55
Your guess is too low
58
Yourguessed the number in 5 tries!
```

```
Hello, What's your name?Azam
okay! Azam I am Guessing a number between 1 and 100:
50
Your guess is too low
75
Your guess is too low
90
Your guess is too high
85
Your guess is too high
80
Your guess is too high
You did not guess the number, The number was 78
```

13. The following code is supposed to open a file 'abc.txt' and counts the number of occurrences of the character 's' and displays output like:

```
1   file_name = 'abc.txt'
2   target_letter = 'S'
3   try:
4       file = open(file_name, "r")
5       text = file.read()
6       file.close
7   except FileNotFoundError:
8       print(f"Error: File '{file_name}' not found.")
9       exit()
10  count = 0
11  for char in text:
12      if char = target_letter:
13          count += 1
14  print("The letter '{target_letter}' appears " + count + " times in the file '{file_name}'.")
```

- Sample text in file 'abc.txt': I scream, you scream, we all scream for ice cream.
- Sample Output: The letter 's' appears 3 times in the file 'abc.txt'.

However, the code does not execute. Debug and remove any bug(s) from the code.

Mini Project 1

Design a game where multiple objects are falling from the top. The objective is to "Catch The Falling Objects" using arrow keys. For every catch, a score increases to 1 point. The game ends whenever an object is missed.

Mini Project 2

Design two textboxes such that the first is for entering text while the second shows the text converted to uppercase. The user can click on the first textbox and type a sentence. As the user types, the second textbox automatically displays the uppercase version. The second textbox is not editable.

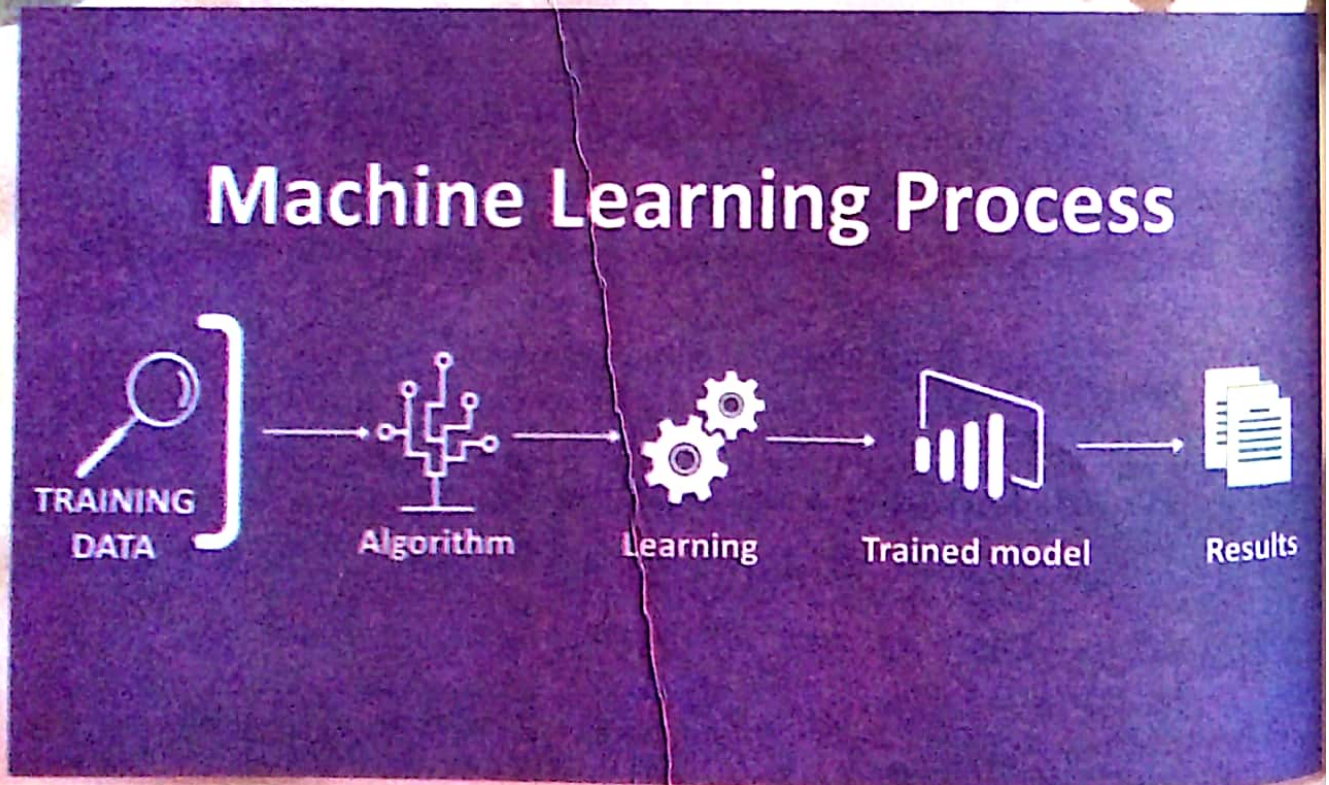
Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

4 Data and Analysis

Learning Outcomes

At the end of this unit students will be able to:

- analyze data and identify key model performance metrics of real-world machine learning models.
- explain and create a data visualization using data visualization software (for example, Structured Query Language (SQL), Python, or R)
- learn how to form hypotheses and perform hypothesis testing. Students will learn to communicate findings using advanced data visuals and tie them back to hypotheses.



Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Introduction

Machine Learning is a method in which computers learn from existing data to make new decisions. A computer learns from the predefined rules, which are provided at the time of training it. These algorithms are designed carefully with the help of raw data and then transforming it into useful models. These models are assessed based on their performance in various environments. During these assessments, the models learn patterns and improve their performance. After attaining a certain level of accuracy, these models are able to predict future values based on the past data patterns. After analyzing the results obtained by such machine learning models, we can understand cause and effect relationships in data. In this way, we can use data and computers to visualize the future results based on existing data

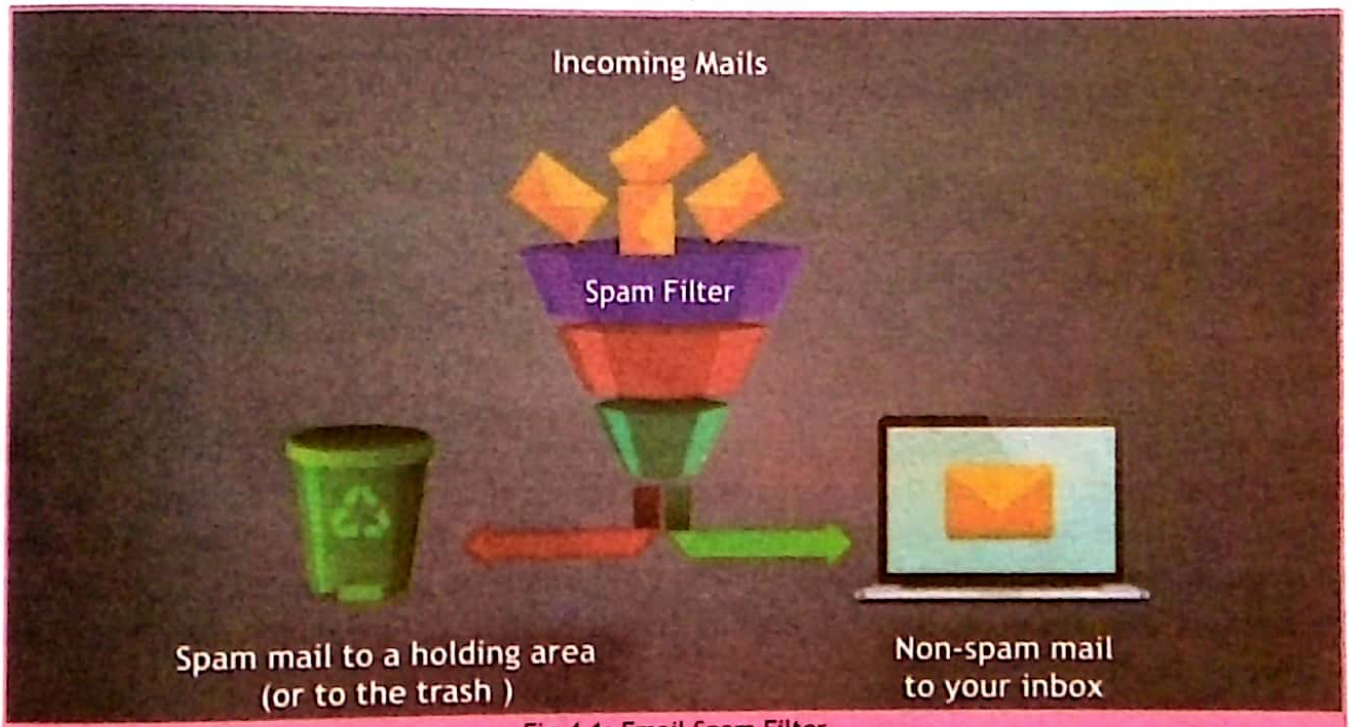


Fig.4.1: Email Spam Filter

We can understand it with the help of an example of Email Spam Filter. A spam filter learns from past emails' labels whether they are spam or not. When we mark some email in our inbox as spam the machine learning model learns from the keywords used in that email. Next time if we receive an email that contains similar words, which were used in "previously marked spam" email, it will automatically send that email into spam folder.

4.1 Data Types

Data is a collection of raw facts and figures collected from different sources. In data science and machine learning data is categorized into various types. The following table provides a brief description and example of some data types:

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Data Type	Description	Example
Numerical	It is quantitative data and used to represent numerical values.	Marks, Salary, Age
Categorical	It is data which has discrete value. It is further divided into Nominal data and Ordinal Data.	Nominal: Male, Female, Color Categorical: Rank, High, Low, Small, Large.
Boolean	It has only two possible values 0 or 1.	True/False, Yes/No
Text Data	It contains a sequence of characters.	Description, Review, Feedback
Time Series	It is collected over a long period of time.	Stock price, weather data
Image	It is represented as a pixel value in the image.	X-Ray, Ultrasound, Photographs
Audio	It is in the form of sound waves.	Speech, Music
Structured	It is organized in the form of rows and columns (tables).	Spreadsheet, Database
Semi Structured	It is organized in some specific formats, but there is no fixed pattern/schema.	XML, JSON files
Unstructured	It is without any predefined structure.	Videos, Emails, Web Pages.

Introduction to Machine Learning

Before going into the core concepts of machine learning, it is important to understand the foundational topics that make these systems work. These include understanding how models are trained, evaluated, and improved using data, as well as learning the key metrics used to measure the performance of these models. The following are some basic but important concepts about Machine Learning. These are essential building blocks to understand Machine Learning models and related topics.

4.1.1 Difference Between Machine Learning and Rule-based Algorithms

Machine learning models learn from data to make decisions. They adapt and improve over time as they process more data. Whereas rule-based algorithms follow fixed instructions. They do not change unless we deliberately update them. Machine learning can handle complex problems and unexpected situations. Rule-based systems are simpler and only work well if the rules are correct. Machine learning models are flexible and can learn new things. Rule-based systems are consistent but need new rules for different problems. Machine learning is more dynamic and can handle a variety of data.

Both Machine Learning and Rule Based algorithms have their advantages and limitations, which is as follows:

Advantages of Machine Learning:

It can handle complex and unstructured data. It can automatically learn patterns from data without any explicit programming. It does not require expert knowledge of human intervention. Its accuracy depends on the size of the dataset.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Advantages of Rule-based Algorithms:

It does not require large datasets. It can work effectively with limited data because it depends upon predefined rules. Its performance is consistent with the same set of rules. It requires fewer resources and can be implemented easily.

Limitations of Machine Learning:

It does not perform well if data is too little. Their decision-making process is complex and cannot be easily interpreted. It requires significant computational resources. If the model is too complex it can lead to overfitting. If training data is biased it can provide wrong results.

Limitations of Rule-based Algorithms:

It has limited flexibility, as it is not adaptable to new data which is not anticipated in predefined rules. It can provide wrong results if data is unstructured or ambiguous. It cannot learn from data. If too many rules accumulate it can slow down the system.

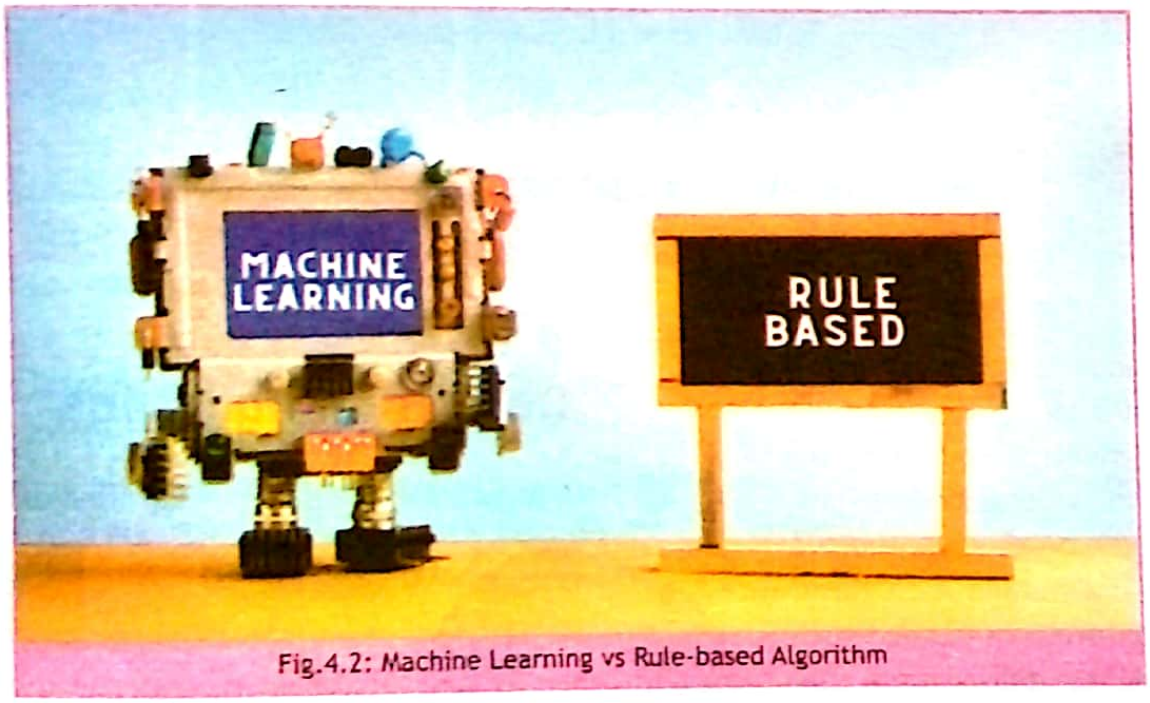


Fig.4.2: Machine Learning vs Rule-based Algorithm

4.1.2 Model Building in Machine Learning

Model building in machine learning involves several key steps to develop a system that can learn from data and make predictions or decisions. At first, data is collected and preprocessed to ensure it is clean and suitable for analysis. This might involve normalizing values, handling missing data, and encoding categorical variables. After data preprocessing, a suitable machine learning algorithm is chosen based on the problem at hand, whether it's classification, regression, clustering, or another type. The algorithm is then trained on the dataset, adjusting its parameters to minimize error and improve accuracy.

During training, the model's performance is evaluated using a separate validation set of data to ensure its accuracy about new unseen data. Once the model is optimized, it is tested on a final test dataset to assess its effectiveness. If the results are satisfactory, the model can be deployed for real-world use; otherwise, it may require further improvements in code. Throughout this

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

process, every time the model reads data and calculates results, the algorithm keeps on learning. Repeated refinement and evaluation help in achieving a model that performs well and meets the specific needs of the application.

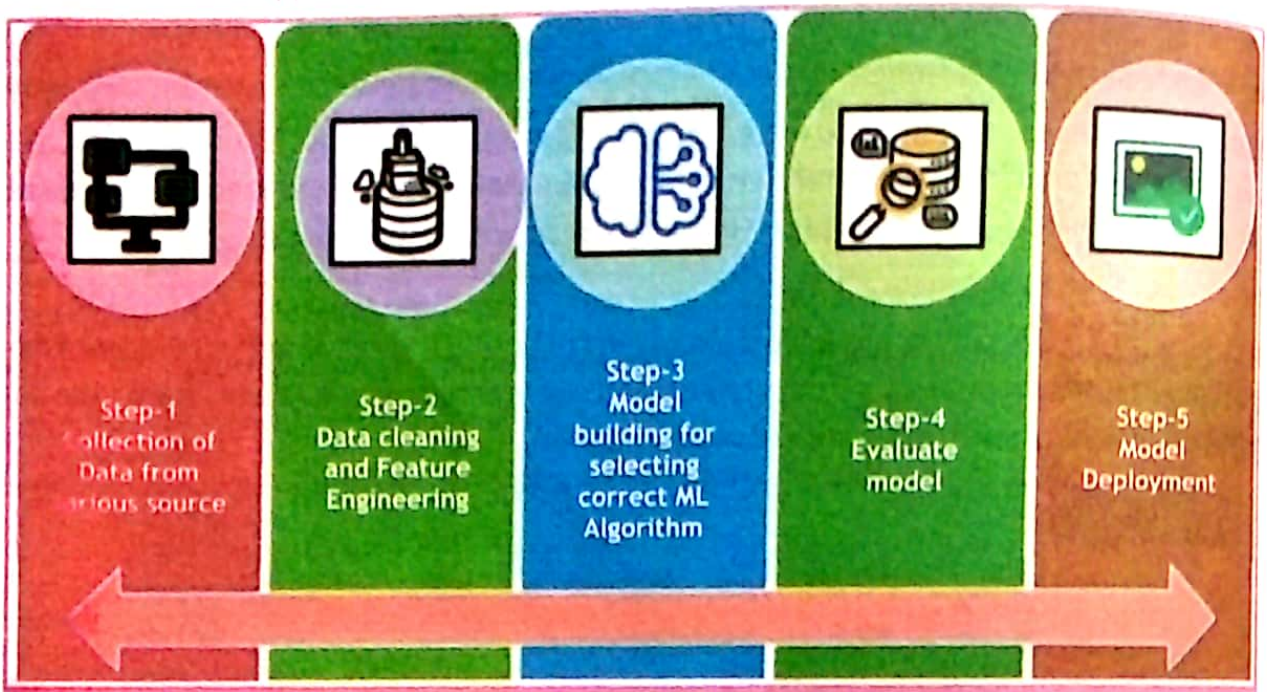


Fig.4.3: Machine Learning Model Building Process

Key Terms in Model Building using Machine Learning:

To understand the entire process of model building using Machine Learning, some key terms are necessary to understand. We will take a real-life problem, so that we can better understand each term. We want to develop a model which can identify or differentiate the images of dogs, cats and horses. We must have a reasonable collection of images of all the three animals. Now we will learn each key term with reference to this example as following:

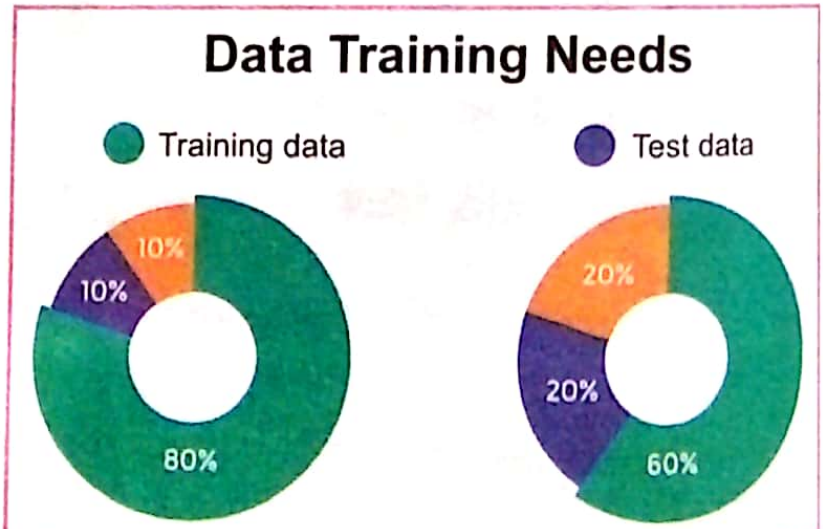


Fig.4.4: Data Training Needs

Feature Engineering: Feature engineering is the process of selecting and transforming the most relevant data features to improve a model's performance. For example, if you want to predict that a given image is of a dog, cat or horse, we have to make a list of common and distinct features of each animal i.e. size, color, appearance, shape of eyes, ears, hair, nose etc. Our model will keep data about all possible variation of features of these animals, so that a sitting horse cannot be confused with the image of a big size standing dog.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Test-Train Split: A test-train split divides data into two parts i.e training data (main portion) and test data (smaller portion). In our example we decided to make a test train split of seventy and thirty percent. This ratio should be chosen wisely. If we provide too much data at the time of training our model will overfit. If we provide too little data for training, then the model will underfit.

Model Training: Model training is the key feature in Machine Learning. It helps the model to make decisions which are more accurate and meaningful. Without proper model training, the machine is unable to decide about the new data. Supervised Learning, Unsupervised Learning and Reinforcement Learning are commonly used Machine Learning models.

In **supervised learning**, the model is trained using labeled data, where the correct output is provided for each input. A common example of a supervised learning algorithm is Linear Regression, which is used to predict continuous values such as prices or temperatures.

In **unsupervised learning**, the data is unlabeled, and the model tries to find patterns or groupings on its own. An example of this type is K-Means Clustering, which groups similar data points together, such as in customer segmentation.

Reinforcement learning involves a model that learns by interacting with an environment, receiving feedback in the form of rewards or penalties. A popular algorithm in this category is Q-Learning, often used in game playing or robotic navigation where the model learns to make decisions to maximize its rewards.

Overfitting: It happens when a machine learning model learns the training data too well, including its noise and irrelevant details. This makes the model perform very well on the training data but poorly on new, unseen data because it's too specific to the training set.

Underfitting: It occurs when a model is too simple and fails to capture the underlying patterns in the data. As a result, it performs poorly both on the training data and on new data, as it hasn't learned enough from the available information.

Model Assessment: A Model assessment evaluates a model's performance using parameters like accuracy, precision, and recall. Keeping in view our example, we provide 100 new images of cats, dogs and horses which were not part of our train and test data. If our model predicts the correct results for 80 images and for rest of 20 images it predicts wrong labels, like it gives horse label for an image of dog then the accuracy of our Machine learning model is 80%.

Accuracy: This is the percentage of total predictions that are correct. It tells how often the model is right overall.

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

Precision: This measures how many of the predictions the model made for a certain class (e.g., "positive" cases) are actually correct. It's about the quality of the positive predictions.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Recall: This tells how well the model identifies all the actual positive cases in the data. It shows the model's ability to find all the relevant instances.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}}$$

True Positive (TP): Model predicts "Cat," and the actual animal is a Cat.

False Positive (FP): Model predicts "Cat," but the actual animal is a Dog or Horse.

True Negative (TN): Model predicts "Not Cat," and the actual animal is a Dog or Horse.

False Negative (FN): Model predicts "Not Cat," but the actual animal is a Cat.

F1 Score: It is the harmonic mean of Precision and Recall. It balances the trade-off between precision and recall. It is very useful when the dataset is imbalanced, giving a single score that reflects both precision and recall performance.

$$F1_{Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Training Data: Training data is the main dataset used to teach a model to make predictions or decisions. In our example we will provide a reasonable number of images of dogs, cats and horses from our huge dataset. If we have a total of one hundred thousand images, then we can provide sixty to seventy thousand images to our Machine Learning Model to learn. Each picture will contain relevant labels like cat, dog or horse if we are using supervised learning technique. These concepts have been already discussed in Grade 11.

Test Data: Test data is a small portion of data that we decide to use for evaluation purposes of our Machine Learning model. For example, after providing training data when our model is trained, we can provide unknow images from our dataset and check whether the model provides the right answer or not. These images will not be given to our model at the stage of training, rather they will be from that forty or thirty percent of images which were not provided at the time of training.

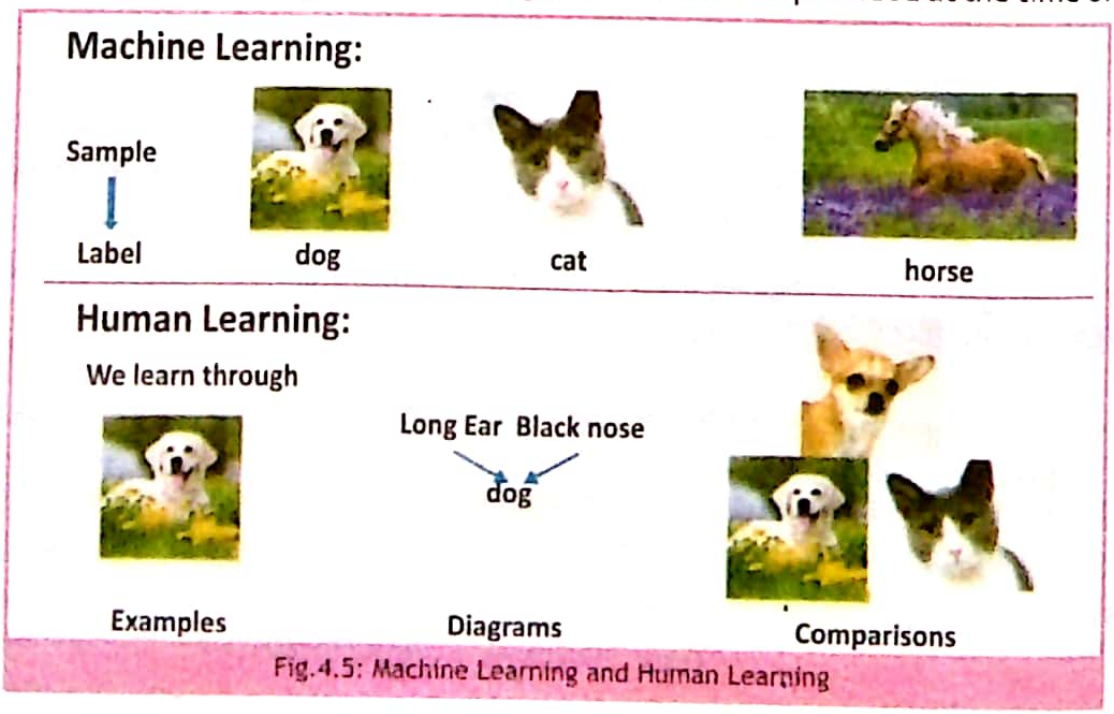


Fig.4.5: Machine Learning and Human Learning

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

4.1.3 Learning from Data

In machine learning, models find patterns in data. The more data the model has, the better it can learn. These models improve their performance based on feedback from the data. Learning from data helps the model make accurate predictions. Different algorithms work better with specific types of data. Learning involves understanding trends and patterns in the data. As the model sees more examples, its performance improves. Data quality affects how well the model learns. For example, if our dataset contains very clear images of cats, dogs and horses the learning of our model will be good. Continuous learning keeps the model accurate and improves its accuracy over time. Good data helps models learn and make better predictions.

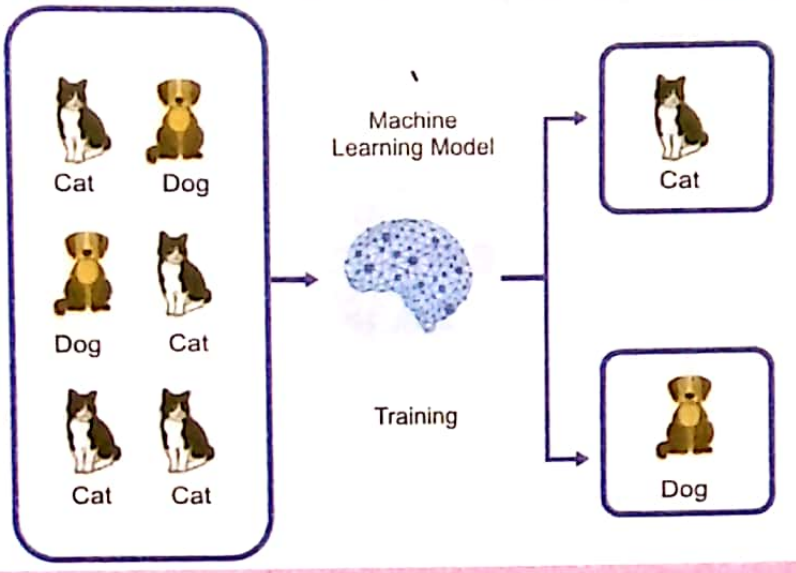


Fig.4.6: Machine Learning Model

4.1.4 Algorithms for Different Data Structures

Selecting appropriate algorithm for Machine Learning model is very important. There are certain guidelines which help us to decide which algorithm we should select. There are numerous algorithms available. It is important to understand which algorithm is more appropriate and provide desired results. The following are some key concepts to consider while choosing the right algorithm:

Algorithm selection according to data type:

1. **Structured data:** It is organized in tables with rows and columns. The structured data is easily searchable and quantifiable. The examples of structured data are databases, spreadsheets, and CSV files. Traditional algorithms like regression and classification are suitable for this type of data.
2. **Unstructured data:** it is unorganized data and has no specific structure. It includes text, images, audio, and video files. The unstructured data lacks a predefined format. The example of unstructured data are social media posts, emails, and sensor readings. Advanced algorithms like neural networks are suitable for this data.

Supervised learning uses labeled data to train models. Unsupervised learning finds patterns in unlabeled data. Regression algorithms predict continuous values like prices. Classification algorithms sort data into categories. Clustering algorithms group similar data points together. Neural networks are used for complex patterns and large data. For example, if the data is a collection of cat images, then there will be very complex patterns like colors, shapes, textures, position of the cat, angle of the picture etc. Neural networks can handle this complex pattern

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

easily. Similarly, if size of our dataset is very large, for example we take social media posts on the topic of COVID-19, it will have a huge number of posts. Neural networks can be trained on many posts in parallel by using powerful computers and GPUs. Therefore, these algorithms are suitable for such tasks. Choosing the right algorithm depends on the data type and problem. Matching algorithms to data structures is key for effective modeling.

4.2 Data Visualization

We are already familiar with the term data visualization in which we used programming paradigms and Python programming language to visualize Tips dataset. Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data. Additionally, it provides an excellent way for employees or business owners to present data to non-technical audiences without confusion.

There are many tools that can help us turn data into pictures and charts, making it easier to understand. For example, SQL is a language that helps us get data from databases. Python has libraries like Matplotlib and Seaborn that create charts and graphs. R is a special tool that's great at creating statistical graphics and visualizations.

Using these tools, we can create charts and graphs that show us patterns and trends in the data. Different tools are better suited for different tasks and skill levels. The right tool can help us present our findings in a clear and effective way. Some tools even let us interact with the data, making it more fun to explore. Choosing the right tool depends on what we want to do with the data. Good visualization tools help us tell a story with the data, making it easier for others to understand.

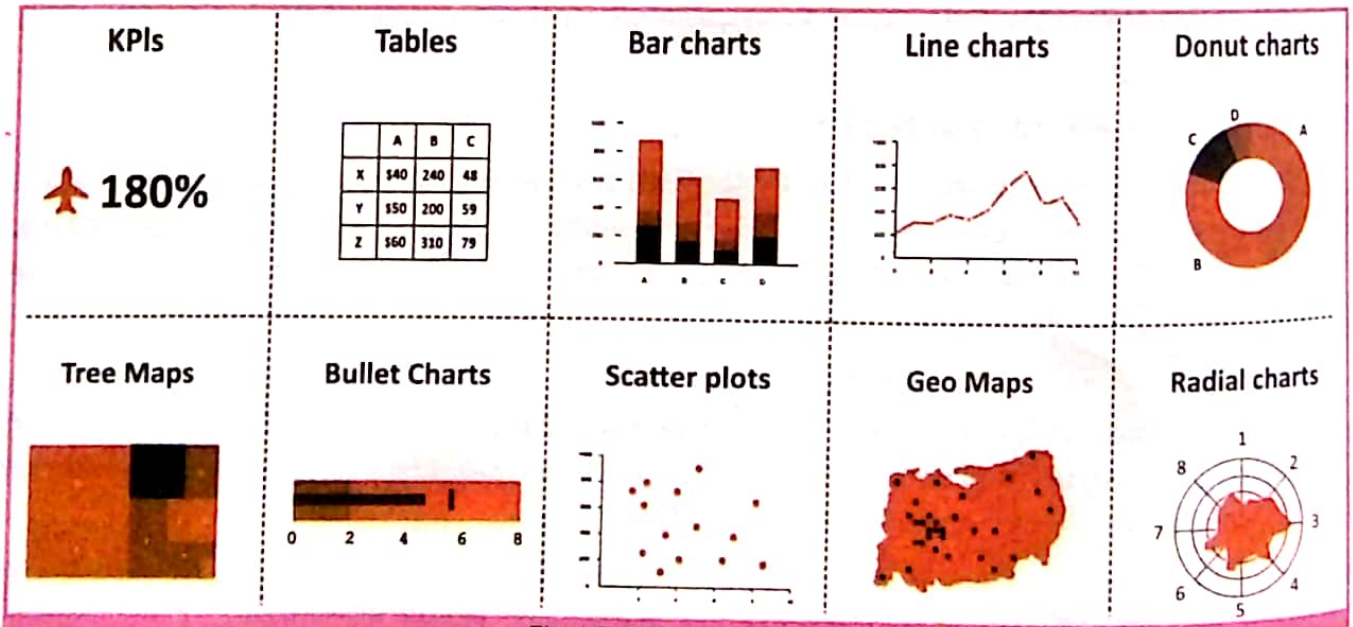


Fig.4.7: Data Visualization

4.2.1 Data visualization by using Python

The following code can be used to learn how python libraries can be used to visualize data:

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

DataVisualization.ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text

```
# Import numpy library and give it a short name 'np'
# We use numpy to create and manage arrays (tables of numbers)
import numpy as np

# Import the 'pyplot' part of matplotlib and give it a short name 'plt'
# We use 'pyplot' to make graphs like scatter plots and lines
import matplotlib.pyplot as plt

# Import the LinearRegression class from sklearn's linear_model module
# We use it to create a machine learning model that can draw a straight line through points
from sklearn.linear_model import LinearRegression

# Step 1: Prepare the data (study hours and marks)
x = np.array([[1], [2], [3], [4], [5]])  # Input: hours studied
y = np.array([10, 20, 30, 40, 50])       # Output: marks obtained

# Step 2: Create and train the Linear Regression model
model = LinearRegression()
model.fit(X, y)

# Step 3: Predict the marks based on study hours
y_pred = model.predict(X)

# Step 4: Plot the actual points and the prediction line
plt.scatter(X, y, color='blue')          # Plot real data points in blue
plt.plot(X, y_pred, color='red')         # Plot the prediction line in red
plt.title('Study Hours vs Marks')        # Title of the graph
plt.xlabel('Hours Studied')              # X-axis label
plt.ylabel('Marks Obtained')             # Y-axis label
plt.show()                               # Show the final graph
```

Fig.4.8: Python Code for Data Visualization

After executing the Python code by using google colab, the following will be the output:

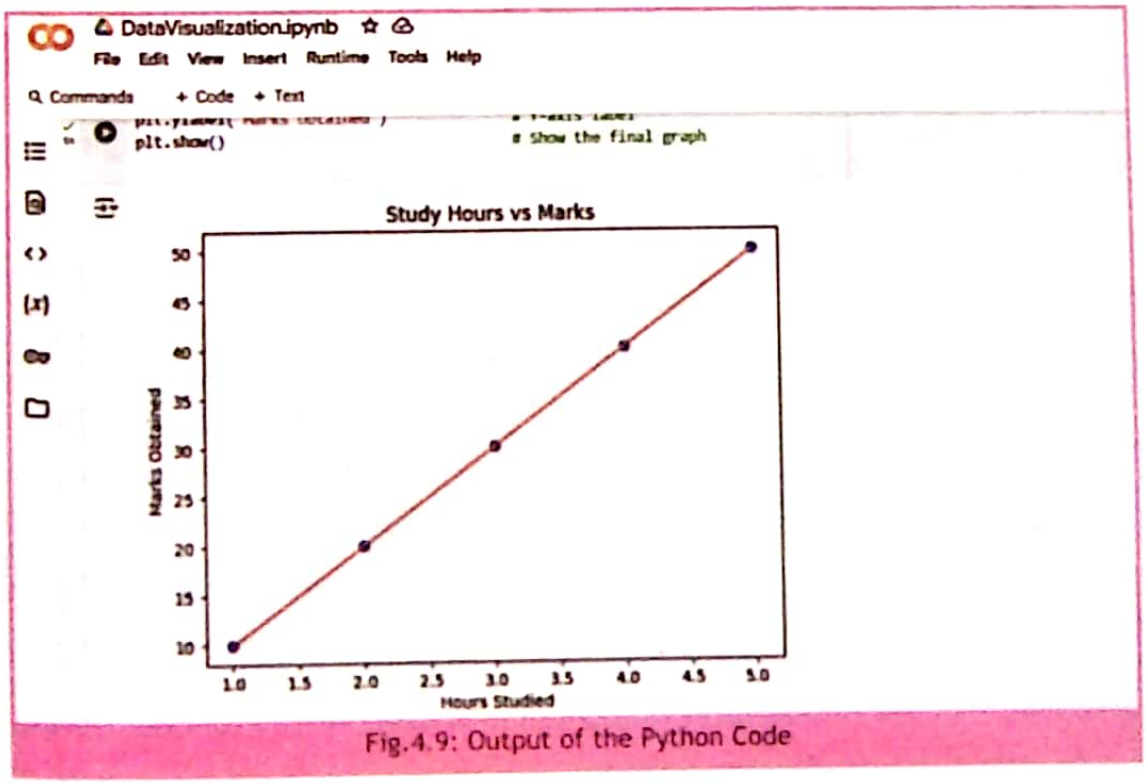


Fig.4.9: Output of the Python Code

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

4.2.2 Data Storytelling

Data storytelling uses visuals to communicate insights from data. It combines charts, graphs, and narratives to explain findings. A good data story engages the audience and makes data relatable. It helps in making data-driven decisions by highlighting key points. Clear storytelling simplifies complex data for better understanding. It shows the impact of the data on real-world issues. Effective data stories make the information memorable and actionable. They provide context and meaning to raw data. Data storytelling bridges the gap between data and decision-making. It turns data into a compelling narrative.

Data Storytelling Example/Case Study

Context: A school wants to improve student attendance.

Data:

- Average attendance rate: 85%
- Highest attendance rate: 9th grade (90%)
- Lowest attendance rate: 12th grade (78%)
- Average absences per student: 5 days

Story:

Meet Jameel, a 12th-grade student who represents our school's attendance challenge. Jameel misses an average of 5 days of school, which puts him at risk of falling behind. But our 9th-grade students are showing us that it is possible to do better, with an impressive 90% attendance rate. What if we could identify the factors driving their success and apply them to our 12th-grade students?

By doing so, we could help Jameel and his fellows to stay on track and do better. Let's work together to make attendance a priority and support our students' success!

Key elements:

1. Relatable character (Jameel)
2. Contextual data (attendance rates, absences)
3. Insight (9th-grade success)
4. Call to action (apply insights to 12th-grade students)

This simple story uses data to paint a picture, evoke empathy, and inspire action. It's a basic example, but it illustrates how data storytelling can make complex information more engaging and memorable.

4.2.3 Formulating Questions and Data Evaluation

Formulating questions guides data exploration and analysis. Identify relevant data sets to answer these questions. Evaluate how well the data addresses the questions posed. It compares new data with previous findings to see if they align or not. The questions are used to focus on specific data insights and trends. They are refined after evaluation to ensure the data quality. The good quality

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

of questions ensures accurate results. Effective question formulation drives meaningful analysis. Good questions lead to better data insights. Data evaluation helps in understanding and interpreting findings. We can better understand by the following example that how to formulate a question and evaluate data:

Topic: Analyzing the Impact of School Breakfast Programs on Student Attendance

Formulating Questions:

1. Does participating in a school breakfast program improve student attendance?
2. Which grade levels benefit most from breakfast programs in terms of attendance?
3. Do students from low-income families show greater improvement in attendance compared to their peers?

Identifying Relevant Data Sets:

1. Student attendance records
2. Breakfast program participation data
3. Demographic data (grade level, family income, etc.)

Fig.4.10: School Breakfast Program

Evaluating Data Quality:

1. Check for missing or inconsistent data
2. Ensure attendance records are accurate and up-to-date.
3. Verify breakfast program participation data matches student records

This example demonstrates how formulating questions guides data exploration and analysis, leading to meaningful insights and informed decision-making.

4.2.4 Descriptive Statistics Techniques

Ther are some descriptive statistical techniques to summarize and describe data. The basic techniques include Mean, Median and Mode. Mean is the average value of the data set. Median is the middle value when data is ordered. Mode is the most frequent value in the data. Range shows the difference between the highest and lowest values. Standard

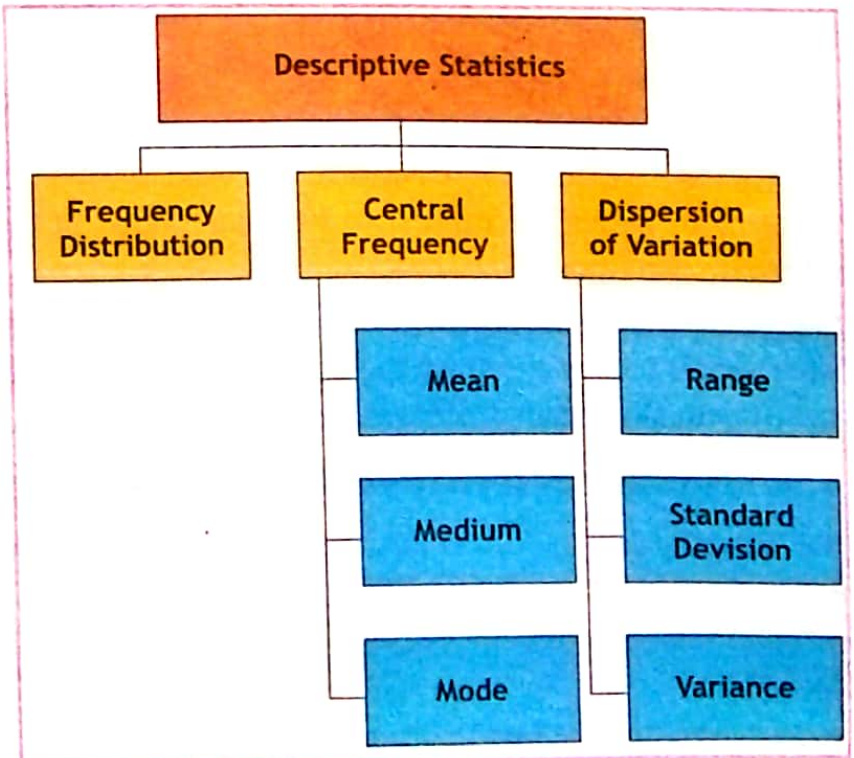


Fig.4.11: Descriptive Statistics Techniques

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Deviation measures how spread out the values are. Variance is the average of squared differences from the mean. Percentiles indicate values below which a certain percentage of data falls. Histograms display the frequency distribution of data. These techniques help in understanding and summarizing data effectively. These statistical techniques are frequently used to get true insight and make informed decisions in the Machine Learning Model.

4.3 Hypothesis Formulation and Hypothesis Testing

Hypothesis formulation is the process of making an educated guess or assumption about a relationship between variables based on existing data or prior knowledge. This assumption, or hypothesis, is a starting point for investigating how certain factors might influence an outcome. It provides a framework for testing and discovering whether these relationships exist in the data or not.

For example, we formulate a hypothesis: "Increased study hours lead to better exam scores for students." This hypothesis proposes a cause-and-effect relationship between study hours (independent variable) and exam scores (dependent variable). A cause-and-effect relationship is the connection between two events or factors, where one event (the cause) directly influences or leads to another event (the effect).

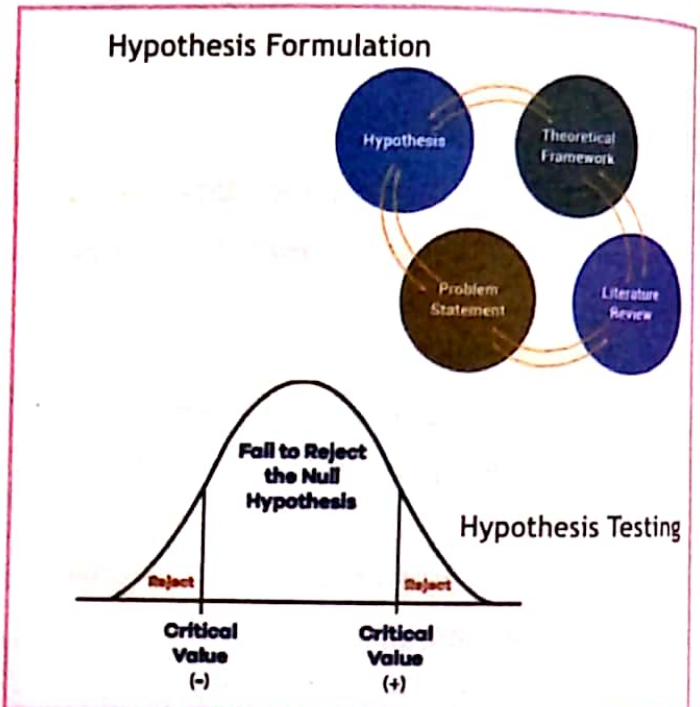


Fig.4.12: Hypothesis Formulation and Hypothesis Testing

Once a hypothesis is formulated, hypothesis testing is the statistical process of determining whether the data supports or refutes the hypothesis. It is a critical tool used to validate assumptions, model performance, and patterns in data. There are two key hypotheses in hypothesis testing, Null Hypothesis and Alternate Hypothesis.

In hypothesis testing, statistical tools like p-values and significance tests are used to determine whether to accept or reject the null hypothesis. It allows us to validate whether a certain is statistically significant or merely due to chance.

4.3.1 Null and Alternative Hypotheses

Null Hypothesis (H_0): This states that there is no effect or relationship between the variables. It assumes any observed differences are due to random chance.

Alternative Hypothesis (H_1): This suggests there is a significant effect or relationship between variables.

The Null Hypothesis assumes no effect or relationship exists. The Alternative Hypothesis suggests

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

an effect or relationship is present. Testing aims to confirm or reject the null hypothesis. The null hypothesis is the default assumption that needs to be challenged. The alternative hypothesis is what researchers want to prove. Results help in deciding which hypothesis is supported. Both hypotheses guide the research and testing process. Clear definitions of these hypotheses are essential for accurate testing. They provide a framework for analyzing data and drawing conclusions. Effective hypothesis testing relies on understanding both types. The following example helps to understand the concept easily:

Fig.4.13: Null & Alternate Hypothesis

Example: "Does drinking coffee improve cognitive function in students?"

Null Hypothesis (H_0): Drinking coffee has no effect on cognitive function in students. (Assumes any observed differences are due to random chance.)

Alternative Hypothesis (H_1): Drinking coffee improves cognitive function in students. (Suggests a significant effect or relationship exists.)

Testing: Researchers collect data on students' cognitive function with and without coffee consumption.

Results: If the data shows a significant improvement in cognitive function with coffee consumption, the null hypothesis is rejected, and the alternative hypothesis is supported.

Conclusion: Drinking coffee has a positive effect on cognitive function in students.

4.3.2 P-values

P-values measure how likely results are due to chance. A low P-value means strong evidence against the null hypothesis. A high P-value suggests weak evidence and possibly random results. Understanding P-values helps in interpreting test results. It ensures that conclusions are based on reliable evidence. Fore example if we flip a coin ten times and we get 7 time heads and 3 times tale. We might consider that coin is not fair, or it just happened by chance.

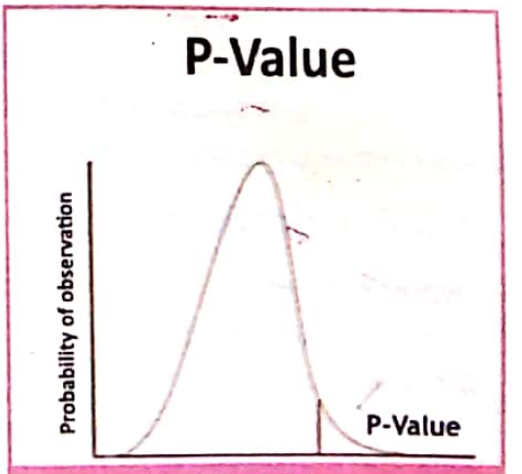


Fig.4.14: P-Values

We can take another example to find the effectiveness of a new fertilizer. We want to see if it helps plants grow taller. We divide the plants into two groups: one group gets the new fertilizer, and the other doesn't. After a month, we measured the height of the plants and found that the fertilized plants grew taller than the others. Now the question arises , "Did the fertilizer really make a difference, or did the plants grow taller just by

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

chance?" The P-value helps answer this. It tells you the chance that the difference in plant height happened randomly, without the fertilizer having any effect.

If the P-value is small (like 0.02), it means it's very unlikely the taller growth happened by chance. This suggests that the fertilizer probably did help the plants grow taller. If the P-value is large (like 0.5), it means there's a good chance the difference in height was just random, and the fertilizer might not have had any real effect. In simple terms, the P-value helps us decide whether the fertilizer really worked or if the result happened by luck. There are certain formulas or tests used to calculate P-values, which are beyond the scope of this grade yet.

4.3.3 Significance Testing

It is a statistical method used to determine if a result is due to chance or it is statistically significant. Statistically significant means it is likely due to some real effect or factor. Significance testing helps determine if results are meaningful. It compares the test statistics with critical value. Setting a significant level helps to decide the threshold for importance. Significance testing helps in making data-driven decisions. It is crucial for validating research findings. For example, if we formulate a research question " Can exercising for six weeks help lower blood pressure?" The null hypothesis is "Exercise doesn't affect blood pressure" while the alternative hypothesis will be "Exercise lowers blood pressure". For statistical experimentation we collect data from the people who exercised for six weeks and had a reduction in blood pressure of 8mmHg. The significant test will be to check if this result is real or just a coincidence. By applying statistical formula, the P-Value will be calculated which indicates that there is only 0.5% chance of getting this result by chance. So, we conclude that since P-Value is very small, that is 0.005 we reject the null hypothesis and accept the alternate hypothesis. It means that exercising for six weeks likely lowers blood pressure.

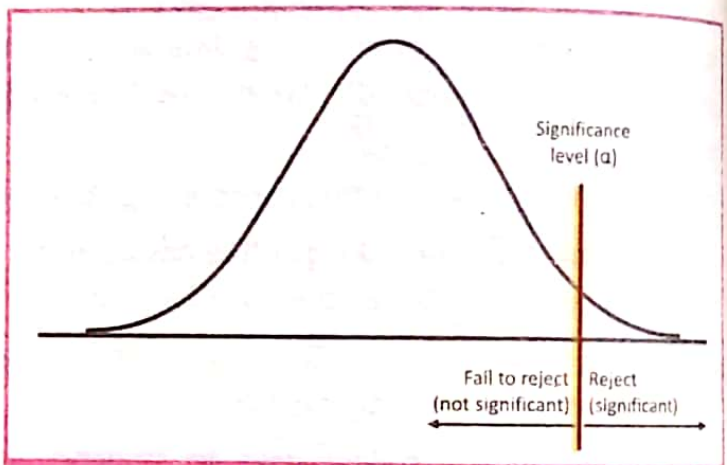


Fig.4.15: Significance Testing

4.3.4 Hypothesis Testing

Hypothesis testing evaluates assumptions using statistical methods. The Test Statistic measures how much data deviates from the null hypothesis. Hypothesis testing involves certain statistical procedures and terms. The terms will be explained with the help of subsequent example. Confidence Interval shows the range where the true value likely falls. Type I Error occurs when a true null hypothesis is incorrectly rejected. Type II Error happens when a false null hypothesis is not rejected. Power measures the ability to detect an effect if one exists. Significance Level is the threshold for determining if results are significant. Sample Size affects the reliability of test results. Understanding these concepts helps in accurate hypothesis testing. They guide in

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

drawing valid conclusions from data. Now we will explore each term briefly with the help of a simple example:

Question: Does drinking coffee improve productivity?

Null Hypothesis (H_0): Coffee has no effect on productivity.

Alternative Hypothesis (H_1): Coffee improves productivity.

Test Statistic: Measure productivity in a sample of people who drink coffee and those who don't. Calculate the test statistic (e.g., t-statistic) to see how much the data deviates from H_0.

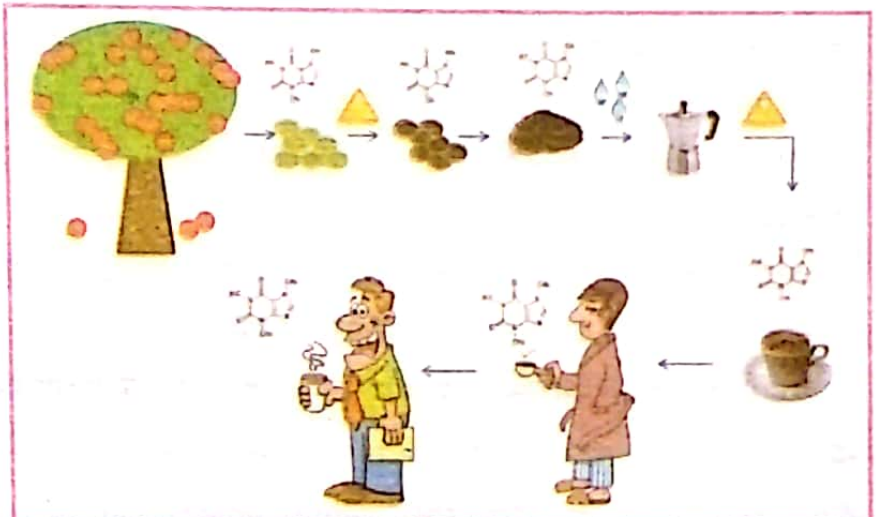


Fig.4.16: Hypothesis Testing

Confidence Interval: Calculate a 95% confidence interval for the average productivity increase with coffee consumption. If the interval is (5, 15), we're 95% confident the true productivity increase falls between 5 and 15.

Type I Error: Rejecting H_0 when it's actually true (i.e., coffee has no effect). This occurs when we conclude coffee improves productivity when it doesn't.

Type II Error: Failing to reject H_0 when it's actually false (i.e., coffee does improve productivity). This occurs when we conclude coffee has no effect when it actually does.

Power: The ability to detect a real productivity increase if coffee actually improves it. A larger sample size increases power.

Significance Level: Set $\alpha = 0.05$ as the threshold for determining if results are significant. If p-value < 0.05, we reject H_0.

Sample Size: A larger sample size (e.g., 1000 people) provides more reliable results than a smaller sample size (e.g., 10 people).

By understanding these concepts, we can accurately test hypotheses and draw valid conclusions from data:

- Hypothesis testing evaluates assumptions using statistical methods.
- Test statistic measures deviation from H_0.
- Confidence interval estimates true values.
- Type I/II Errors occur from incorrect assumptions.
- Power detects real effects.
- Significance level sets thresholds.
- Sample size impacts reliability.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

These concepts help us make informed decisions based on data analysis.

4.3.5 Steps of Simple Hypothesis Testing

Simple hypothesis testing involves a certain sequence of steps. We can list them as follows:

- We start by formulating null and alternative hypotheses.
- Collect data and perform the analysis.
- Calculate the test statistic from the data.
- Compare the test statistic with the critical value to draw conclusions.
- Determine the P-value to assess significance.
- Decide whether to reject or not reject the null hypothesis.
- Use the results to make data-driven decisions.
- Interpret the findings in the context of the research question.
- Report the results clearly with appropriate visuals.

Therefore, hypothesis testing helps in understanding basic data relationships.

4.3.6 Creating Dashboards with Python

A dashboard is a visual tool that helps to understand data by displaying it in the form of charts, graphs, tables etc. We can create these dashboards by using Python programming language. In Python Dash is a framework developed by Plotly to create interactive dashboards. Dash allows us to build dashboards by combining Python's powerful libraries Plotly for interactive charts and Dash for building web layout. The following is a simple code to create a dashboard:

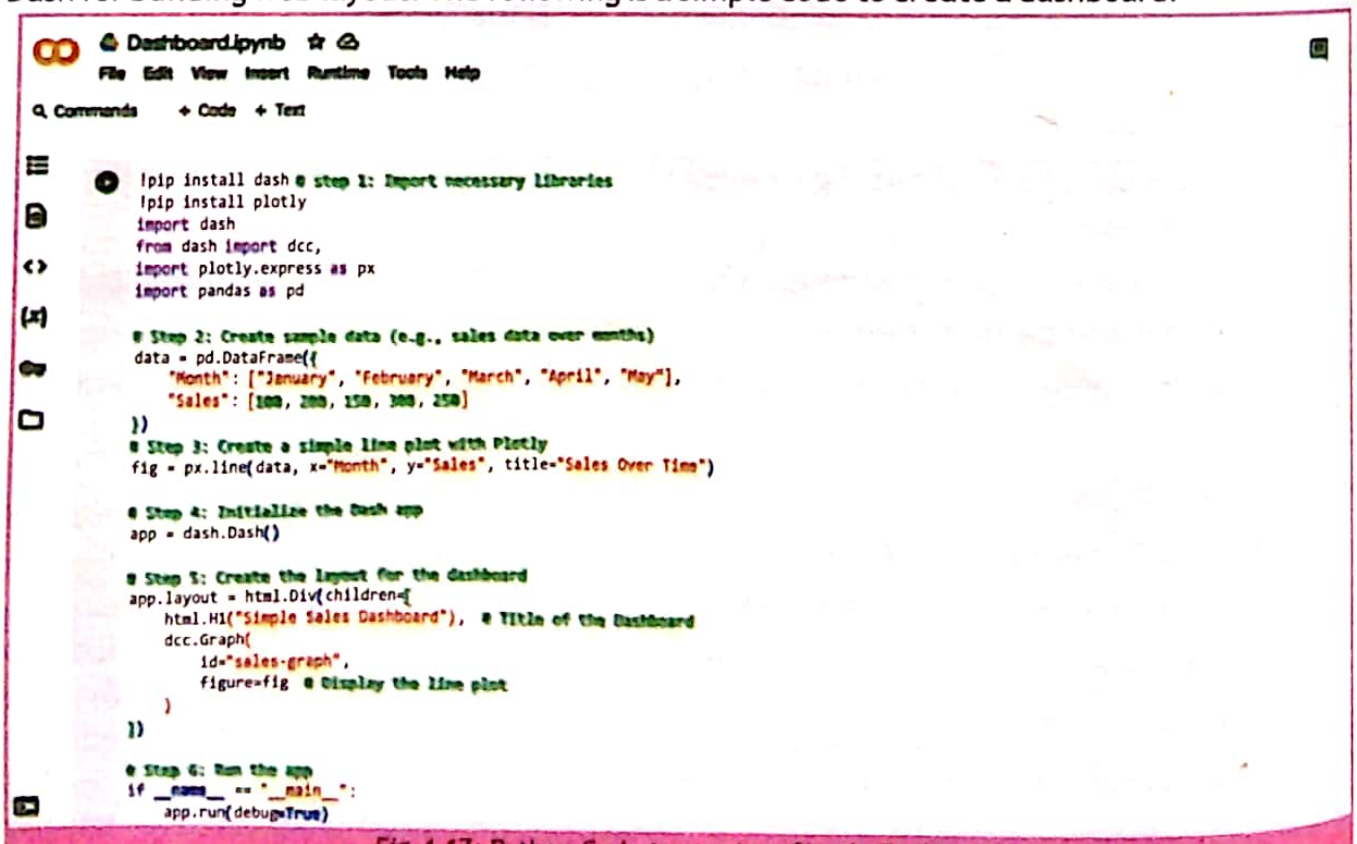

DashboardLipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text
!pip install dash # Step 1: Import necessary libraries
!pip install plotly
import dash
from dash import dcc,
import plotly.express as px
import pandas as pd
# Step 2: Create sample data (e.g., sales data over months)
data = pd.DataFrame({
"Month": ["January", "February", "March", "April", "May"],
"Sales": [100, 200, 150, 300, 250]
})
# Step 3: Create a simple line plot with Plotly
fig = px.line(data, x="Month", y="Sales", title="Sales Over Time")
# Step 4: Initialize the Dash app
app = dash.Dash()
# Step 5: Create the layout for the dashboard
app.layout = html.Div(children=[
html.H1("Simple Sales Dashboard"), # Title of the Dashboard
dcc.Graph(
id="sales-graph",
figure=fig # Display the line plot
)
])
# Step 6: Run the app
if __name__ == "__main__":
app.run(debug=True)

Fig.4.17: Python Code to create a Simple Dashboard

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

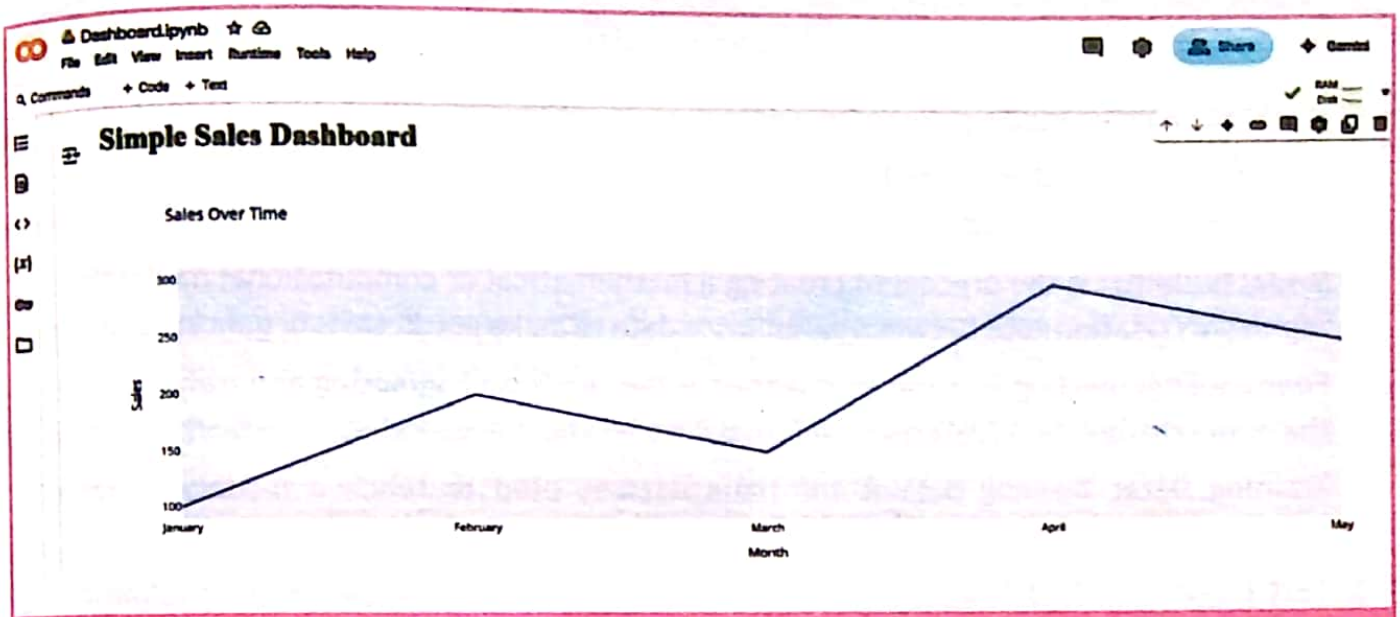


Fig.4.18: Output of the Python Code to create a Simple Dashboard

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Glossary

- **Machine learning models:** learn from data to make decisions.
- **Rule based algorithms:** follow fixed instructions. They do not change unless we deliberately update them.
- **Model building:** is the process of creating a mathematical or computational model that represents relationships between variables in data to make predictions or gain insights.
- **Feature Engineering:** Feature engineering is the process of selecting and transforming the most relevant data features to improve a model's performance.
- **Training Data:** Training data is the main dataset used to teach a model to make predictions or decisions.
- **Test Data:** Test data is a small portion of data that we decide to use for evaluation purposes of our Machine Learning model.
- **Test-Train Split:** A test-train split divides data into two parts i.e training data (main portion) and test data (smaller portion).
- **Overfitting:** It happens when a machine learning model learns the training data too well, including its noise and irrelevant details. This makes the model perform very well on the training data but poorly on new, unseen data because it's too specific to the training set.
- **Underfitting:** It occurs when a model is too simple and fails to capture the underlying patterns in the data. As a result, it performs poorly both on the training data and on new data, as it hasn't learned enough from the available information.
- **Model Assessment:** A Model assessment evaluates a model's performance using parameters like accuracy, precision, and recall.
- **Accuracy:** This is the percentage of total predictions that are correct. It tells how often the model is right overall.
- **Precision:** This measures how many of the predictions the model made for a certain class (e.g., "positive" cases) are actually correct. It's about the quality of the positive predictions.
- **Recall:** This tells how well the model identifies all the actual positive cases in the data. It shows the model's ability to find all the relevant instances.
- **Structured data:** is organized in tables with rows and columns.
- **Unstructured data:** includes text, images, and more.
- **Supervised learning :**uses labeled data to train models.
- **Unsupervised learning:** finds patterns in unlabeled data.
- **Regression algorithms:** predict continuous values like prices.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- **Classification algorithms:** sort data into categories. Clustering algorithms group similar data points together.
- **Neural networks:** are used for complex patterns and large data.
- **Predictive outcomes:** forecast future events based on existing data patterns.
- **Causality:** shows cause-and-effect relationships
- **Model interpretation** helps us understand how a model makes decisions. It explains why the model gives certain predictions.
- **In healthcare:** it predicts disease outbreaks and helps diagnose conditions.
- **In finance:** it detects fraudulent transactions to protect accounts.
- **Retailers** use it to personalize shopping experiences for customers. Transportation companies optimize delivery routes to save time.
- **In agriculture**, it monitors crop health for better yields.
- **Education systems** tailor learning materials to individual needs. Use cases show how machine learning can solve practical problems.
- **Entertainment:** platforms recommend movies or songs based on user preferences. They highlight the impact and benefits of applying machine learning.
- **Data storytelling:** uses visuals to communicate insights from data. It combines charts, graphs, and narratives to explain findings.
- **Reading and critiquing data stories** helps improve understanding. To get a clear understanding of the data story look for clarity and accuracy in published visuals.
- **Clarity of Purpose:** A good data story should have a clear goal or message. The first step is to understand the objective.
- **Data Accuracy and Source Credibility:** The accuracy of the data is fundamental. The sources of the data used in the story should be accurate and credible.
- **Relevance:** Relevance means that the data and narrative should be directly connected to the main topic or issue being discussed.
- **Insight:** refers to the deeper understanding that the data story provides beyond just presenting facts. A good data story should reveal patterns, trends, or conclusions that help the audience learn something new or make better decisions.
- **Persuasion:** Persuasion is about how well the data story influences or convinces the audience to take an action or consider a particular point of view
- **Formulating questions:** guides data exploration and analysis. Identify relevant data sets to answer these questions. Evaluate how well the data addresses the questions posed.
- **Descriptive statistical techniques:** include Mean, Median and Mode. Mean is the

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

average value of the data set. Median is the middle value when data is ordered. Mode is the most frequent value in the data.

- **Hypothesis formulation** is the process of making an educated guess or assumption about a relationship between variables based on existing data or prior knowledge.
- **Null Hypothesis (H_0):** This states that there is no effect or relationship between the variables. It assumes any observed differences are due to random chance.
- **Alternative Hypothesis (H_1):** This suggests there is a significant effect or relationship between variables.
- **P-values** measure how likely results are due to chance. A low P-value means strong evidence against the null hypothesis. A high P-value suggests weak evidence and possibly random results.
- **Significance Testing:** is a statistical method used to determine if a result is due to chance or it is statistically significant. Statistically significant means it is likely due to some real effect or factor.
- **Hypothesis testing:** evaluates assumptions using statistical methods. The Test Statistic measures how much data deviates from the null hypothesis. Hypothesis testing involves certain statistical procedures and terms.
- **Visuals:** help in presenting the results of hypothesis testing. We can use graphs and charts to show data distributions. These visuals can illustrate the test statistics and P-values.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. The spam filter is used to do which of the following tasks:
   a. It helps to delete all emails
   b. It learns from past emails to identify spam emails
   c. It increases the email storage capacity
   d. It mark all the emails to spam
2. Overfitting in a Machine Learning Model leads to:
   a. The model performs poorly on training data
   b. The model performs well on training data but poorly on new data
   c. The model performs equally well on all types of data
   d. The model does not generate results at all
3. The feature engineering involves:
   a. Collecting raw data
   b. Transforming and selecting relevant data features
   c. Testing the model's accuracy
   d. Calculating the significance of the model
4. What is precision in a Machine Learning Model?
   a. The percentage of total correct predictions
   b. The measure of how many positive predictions are correct
   c. The measure of how well the model identifies all actual positive cases
   d. The percentage of training data used
5. How does continuous learning impact a Machine Learning Model?
   a. It reduces the model's accuracy over time
   b. It helps the model improve its performance by adapting to new data
   c. It makes the model slower to process data
   d. It simplifies the model's decision-making process
6. How does storytelling enhance the communication of data insights?
   a. By focusing solely on statistical analysis
   b. By using visuals and narrative to make data relatable and engaging
   c. By ignoring complex data patterns.
   d. By minimizing the use of charts and graphs
7. Which of the following statements describe a Null hypothesis?
   a. It proposes that there is a significant effect or relationship between variables
   b. It assumes that observed differences are due to random chance

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

c. It is the hypothesis that researchers want to disprove

d. It is a statistical method used to measure variability

8. In hypothesis testing P-Value indicates:

a. The probability that the null hypothesis is true

b. The probability that the results are due to some random chance

c. The significance level for the test statistics

d. The confidence interval for the hypothesis

9. Why is it important to use visuals effectively in hypotheses testing?

a. To replace the need for statistical analysis

b. To present data in a more understandable manner and enhance communication

c. To make the data collection process easier

d. To avoid need of a hypothesis formulation

10. In evaluating the credibility of a data story, which factor is least important?

a. Source of data

b. Clarity of the visual presentation

c. The time of day the data was collected

d. Relevance to the topic

11. A model shows 98% accuracy, but precision and recall are both below 60%. What is the most likely issue?

a. The model is underfitting

b. The data is imbalanced

c. The model is overfitting

d. Labels are missing in training data

12. A retail analyst wants to segment customers to discover hidden shopping patterns without any labeled categories. Which approach is most appropriate?

a. Logistic regression

b. Clustering

c. Decision trees

d. Linear regression

Give short answers to the following Short Response Questions (SRQs).

1. Explain what 'Feature Engineering' is and give an example related to image classification.
2. Describe the difference between 'Overfitting' and 'Underfitting' in Machine Learning.
3. What is the purpose of 'Test Data' in evaluating a Machine Learning model?
4. Define 'Model Assessment' and list two metrics used for it.
5. What is the main goal of 'Data Visualization' in data analysis?
6. What is 'Data Storytelling' and why is it important?
7. How does continuous learning benefit a Machine Learning model?
8. Explain the term 'Test-Train Split' and its importance in model training.
9. How does the significance level affect hypothesis testing?
10. Describe how you would identify and mitigate bias in a published data story.
11. What are the key steps in hypothesis testing and why are they important?

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

12. Explain the role of standard deviation in understanding data variability.
13. How can descriptive statistics be used to summarize a large data set effectively?
14. How does the concept of statistical significance support or challenge conclusions in experimental research? Explain with an example.
15. How can data visualization bridge the gap between technical analysis and non-technical audiences? Explain with a real-world scenario.

Give long answers to the following Extended Response Questions (ERQs).

1. Discuss the differences between Machine Learning models and Rule-based Algorithms, including their advantages and limitations.
2. Explain the process of model building in Machine Learning, including data collection, preprocessing, training, and evaluation.
3. Compare and contrast 'Predictive Models' and 'Causal Models,' discussing their uses and limitations.
4. Discuss how formulating good research questions influences the quality of data analysis. Provide an example of how a well-formulated question led to meaningful insights in a study.
5. Examine the process of hypothesis formulation and testing in a real-world context. How do null and alternative hypotheses guide research and decision-making? Illustrate with a relevant example.
6. Critique a published data story for its use of clarity and accuracy. Identify any misleading or biased elements and suggest improvements.
7. Explore the role of p-values in hypothesis testing. How do they help determine the significance of research findings, and what are their limitations?
8. Analyze the steps of simple hypothesis testing and their importance in validating research findings. How does each step contribute to drawing accurate conclusions?
9. Discuss the challenges in ensuring data accuracy and relevance in a data story. How can these challenges be addressed to improve the credibility of the data story?
10. Evaluate how feedback can be used to refine data storytelling techniques. Provide examples of how constructive criticism can lead to more effective data presentations.
11. Discuss the challenges in ensuring data accuracy and relevance in a data story. How can these challenges be addressed to improve the credibility of the data story?
12. Evaluate how feedback can be used to refine data storytelling techniques. Provide examples of how constructive criticism can lead to more effective data presentations.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Activity 1: Feature Engineering Exercise:

Objective: Identify and list features for a given dataset. i.e. images of the animals.

1. Divide the students into small groups depending on the available resources and strength of the class.
2. Helps the students to find some pet datasets from the internet. For example "Cats vs Dogs" dataset from https://www.kaggle.com/c/dogs-vs-cats or
   Oxford pet dataset from https://www.robots.ox.ac.uk/~vgg/data/pets/
3. Ask the students to select relevant features that can help classifying images of dogs, cats and horses.
4. Supervise the students to identify new data and make predictions about the unseen data based on the information learned from the dataset.

Outcome: This activity will help students understand how Machine Learning Models learn data and make predictions about new unseen data.

Activity 2: Data Story Critique.

Objective: Students will practice critiquing data stories by evaluating clarity accuracy insight and persuasion.

1. Divide the students into small groups depending on the strength of the class.
2. Assign or let them choose a data story of their own choice e.g. economic trends, social media usage, prices of smartphones, performance of various video games etc.
3. Each group evaluates their assigned story using the critique framework.
4. The group present their story to the class and provide feedback on how the data story can be improved

Outcome: This activity will help students develop critical thinking, collaboration, communication and data analysis.

Activity 3: Hypothesis Testing Role Play

Objective: To help students simulate hypothesis testing using real world examples.

1. Assign the students various roles like researchers, statisticians, subjects and create a scenario such as "Does increased study hours improve grades?"
2. Students playing the role of researchers will formulate null hypothesis and alternate hypothesis.
3. Students playing the role of statisticians will analyze data provided by the teacher.
4. Students playing the role of subjects will help interpret the results and make decisions based on data.

Outcome: This activity will help students to develop the skills of hypothesis formulation, statistical reasoning and decision making.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Activity 4: Implementation of Machine Learning by designing an interactive dashboard

Objective: To help students understand how to design and create interactive dashboards by simulating a real-world project.

1. Assign students different roles like: Data Collectors (gather and clean the sample data), Dashboard Designers (design how the dashboard will look), Graph Makers (create plots and graphs using Plotly), App Builders (set up the layout and run the Dash app)
2. Create a scenario like, The school principal wants a dashboard to track students' performance across subjects and attendance trends.
3. Students playing the role of Data Collectors will organize basic data, such as students' marks and attendance percentages.
4. Students playing the role of Dashboard Designers will sketch or plan which graphs and filters should be on the dashboard.
5. Students playing the role of Graph Makers will use Plotly to create charts like bar graphs, line charts, and pie charts.
6. Students playing the role of App Builders will use Dash to assemble everything into a working dashboard and run it.

Outcome: This activity will help students develop skills in Data organization, Data visualization, Basic app development, and Collaborative project work. They will also experience how real-world data dashboards are built by teams working together.

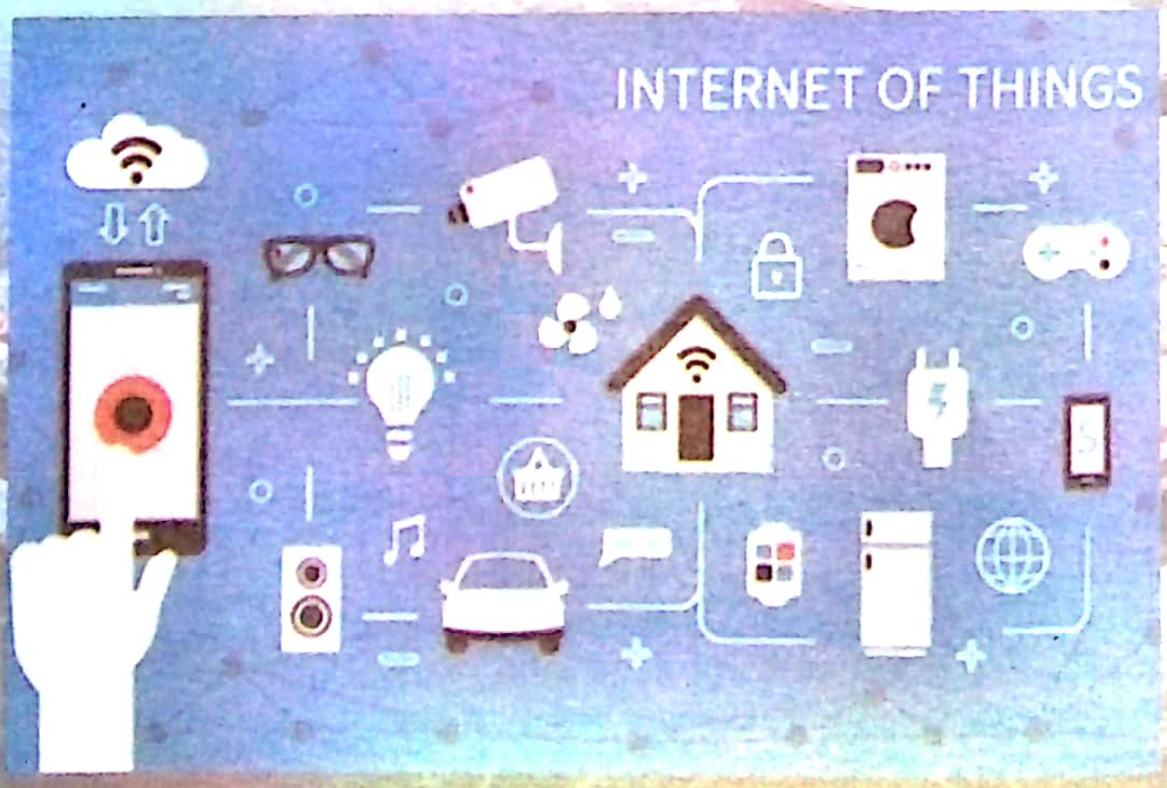
Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5 Applications of Computer Science

Learning Outcomes

At the end of this unit students will be able to:

- design ideas of applications relevant to Pakistan using IoT, Cloud Computing, and Blockchain.
- describe deep learning and its applications.
- assess policies that can help protect different stakeholders' interests.
- evaluate scenarios with data sharing and privacy conflicts and suggest policy decisions that can help achieve acceptable compromises.



Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.1 Internet of Things (IoT)

IoT has big impact on today's modern society. In Pakistan it is at an early stages of planning and implementation. IoT technology improves productivity and efficiency in homes, offices and industries. Application of IoT in government and private sector in Pakistan will enhance our daily life by allowing us to perform daily activities with convenience and comfort. IoT technology provides enhanced automation, control, comfort and economical use of energy. Therefore, it is important to use IoT technology in Pakistan.

5.2 Designing IoT-based Applications

Today, there are billions of Internet-connected devices all overs the world and the number is rapidly increasing. IoT technology provides interconnectivity between various devices, machines and people through Internet that has resulted in innovations in many industries and organizations. Designing an IoT application means incorporating IoT technology into applications for specific purposes.

The following are examples of IoT-based applications.

5.2.1 IoT-based Air Pollution Monitoring

Pakistan is facing serious air pollution in big cities due to industrial emission, vehicle emission, brick kilns and seasonal crop burning. High level of air pollution is causing severe public health concerns such as respiratory and heart diseases.

IoT-based air pollution monitoring system integrated with cloud computing and Blockchain technologies can provide a solution to the environmental calamity of air pollution. It involves the following three stages.

1. IoT-based Real-time Data Collection using Sensors

Deploy a network of IoT sensors across a region having high level of air pollution to measure various pollutants like CO, NO2, etc., and environmental conditions such as temperature, humidity, etc. The sensors collect data and transmit it to a central network hub using wireless communication like Wi-Fi. The network hub sends the data to the cloud computing for storage and processing.

2. Cloud Platform for Data Storage and Processing

Use cloud platform such as AWS or Google Cloud to receive and store the massive amount of data collected by the IoT sensors. Cloud can also store historical data for trend analysis and forecasting. The cloud platform uses analytics techniques such as machine learning and data visualization to analyze and determine Air Quality Indices (AQI) in real-time. When pollution level exceeds certain level it generates alerts and notifications. Based on historical data and pollution patterns, cloud platform can also predict future air quality.

3. Blockchain for Transparency and Security

Create a Blockchain ledger to ensure integrity and authenticity of collected data. Blockchain technology will make the recorded data tamper-proof and less vulnerable to manipulation.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Finally, integrate the IoT sensors, cloud platform and Blockchain network by defining data formats, communication protocols and security measures. Develop user-friendly interface such as web dashboards or mobile apps to visualize air quality data and track trends and receive alerts.

5.2.2 IoT-based Flood Management

Floods are very common in some regions of Pakistan. Rivers and lakes overflow due to excessive rainfall. People lose their lives and there is significant property damage. We need a plan to minimize occurring of floods and damage caused by it by improving forecast, monitoring, and quick response to it.

Integrating IoT, cloud computing and Blockchain technologies can provide a powerful approach to flood management by real-time monitoring, data analysis and secure information sharing. This integrated system will allow for proactive flood risk assessment, early warnings and efficient response planning.

The following is a plan for flood management by making use of IoT-based monitoring integrated with cloud platform and Blockchain technologies.

1. IoT-based Real-time Data Collection using Sensors

IoT sensors can be deployed in flood-prone areas to monitor water levels, rainfall and other relevant parameters such as temperature, humidity, wind speed, wind direction and water temperature in real-time. The collected data is transmitted to cloud platform via wireless communication for analysis and further processing. Real-time data collection and uploading it to cloud platform is the core of the flood monitoring system.

2. Cloud Computing for Data Storage and Processing

Cloud platform provides a secure platform to store, analyze and process large volumes of flood-related data. Machine learning algorithms can be used to identify patterns and predict flood risks. Computer vision and image processing techniques can be used to create detailed maps of flood zones in real-time. When any monitored data exceeds the upper limit, the cloud platform will send alarm information to the administrator for quick response for timely evacuation and preparedness. Cloud-based platforms enable better decision-making and coordination among different agencies.

3. Blockchain for Transparency and Security

Blockchain technology can be used for secure record-keeping. It can create an immutable and transparent record of flood-related events. Blockchain can also be used in tracking the movement of relief materials, monitoring the progress of recovery efforts and ensuring that aid is delivered to those who need it.

Combining IoT, cloud computing and Blockchain for flood management provides proactive planning leading to better safety and reduced damage.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.3 IoT Applications Applicable to Pakistan

5.3.1 Application of IoT in Agriculture

Agriculture is the backbone of Pakistan's economy. As the populations of Pakistan is increasing the demand for food is also increasing. We need to move from traditional agriculture towards smart agriculture based on IoT. It is essential to reshape our agricultural methods by using IoT technology.

The following are the benefits of smart agriculture.

- It allows farmers to have proper knowledge about climate by the use of sensors in IoT devices that observe real-time weather conditions.
- Data collected by sensors such as temperature, humidity, rainfall and soil condition makes farming more controlled, precise and efficient.
- Drones are used for monitoring crop planting, crop health and spraying fertilizer and pesticides and also help for irrigation.
- IoT-based farming will reduce production cost and contribute a lot in economical development.

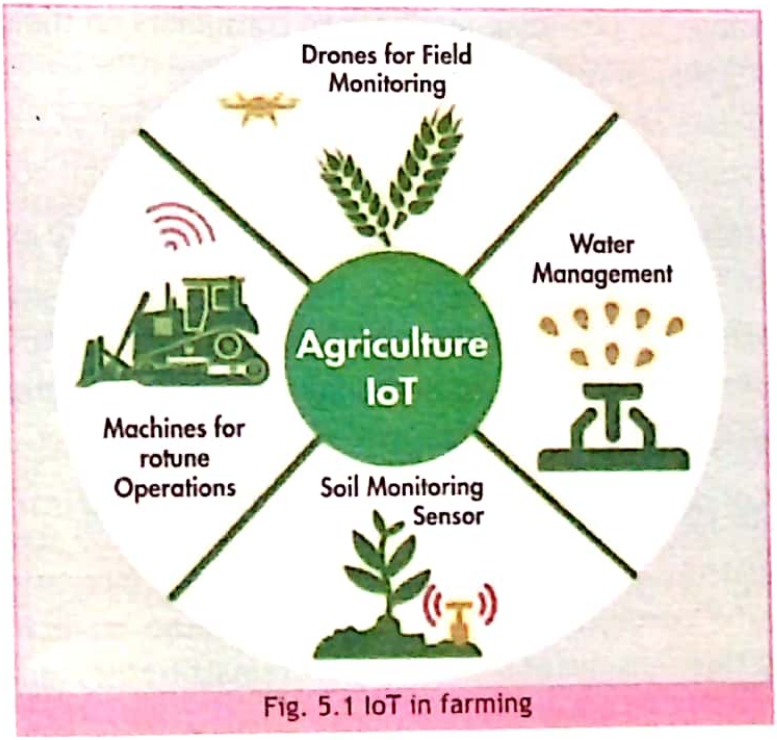


Fig. 5.1 IoT in farming

5.3.2 Application of IoT in Smart Grid

With high cost, shortage and growing requirements of electricity in Pakistan, we need IoT-based smarter solutions to optimize electrical usage and reduce consumer bills. To achieve this government needs to replace the old traditional grids with smart grids.

The infrastructure of electricity supply consists of transmission lines, transformers, substations and power grids. Smart grid also uses the same infrastructure but it involves IoT devices that can communicate with each other and with the consumers.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

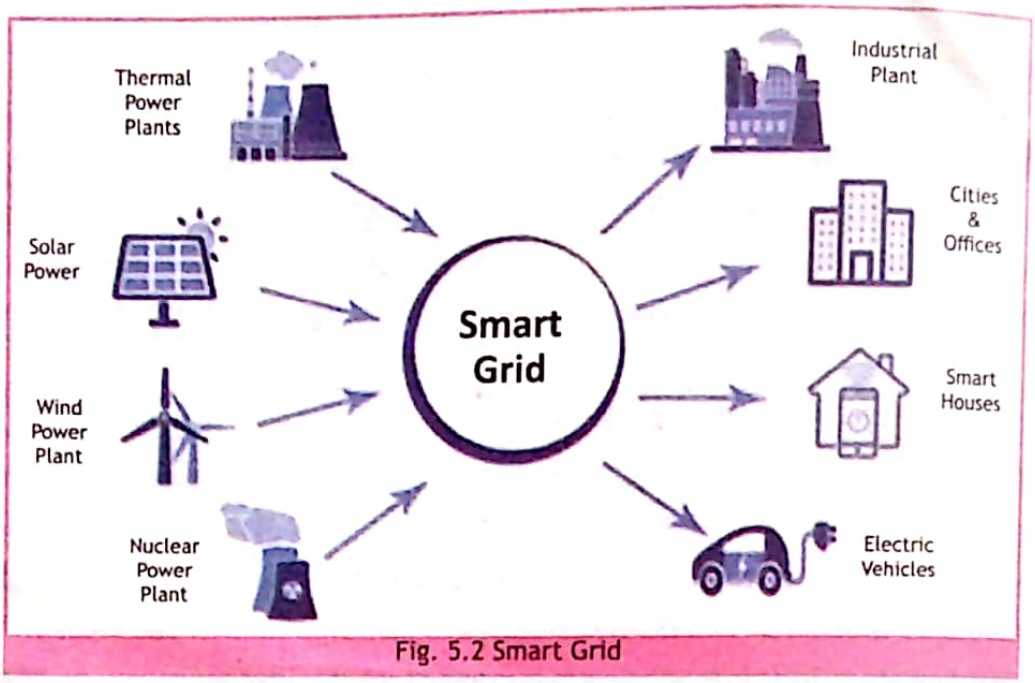


Fig. 5.2 Smart Grid

The following are the benefits of using smart grid.

- It provides more reliable power supply management by having better fault detection.
- It reduces power wastage by providing feedback to consumers on their electrical usage and ensures proper electric supply.
- It analyses the production of electricity and predicts when power shortage can occur so that power companies can take necessary steps to reduce the chances of it occurring.
- It can automatically reroute electricity when there is a power failure avoiding blackout of an entire area.
- It allows power companies to quickly restore power after power failure.
- During peak hours, IoT application suggests consumers to reduce power consumption.
- It generates more accurate bills.

5.3.3 Application of IoT in Traffic Management System

Traffic congestions, particularly in big cities in Pakistan are increasing with the ongoing population growth. Traffic congestions during rush hours cause more fuel consumption and wastage of time as well. This demands implementation of smart traffic management system that uses IoT technology.

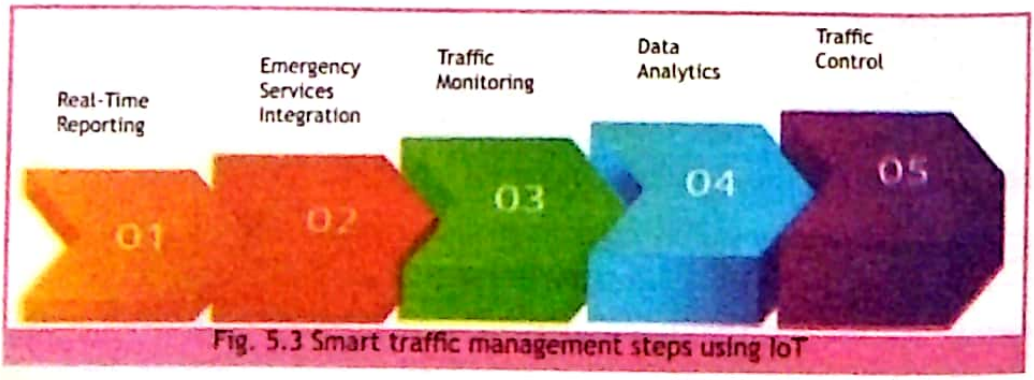


Fig. 5.3 Smart traffic management steps using IoT

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

The following are the benefits of application of IoT in traffic management system.

- Communication through Internet, computer networks and IoT devices enable the implementation of IoT based smart traffic management system.
- Sensors and IoT devices installed at traffic lights, roads and highways wirelessly collect real-time traffic data and send it to traffic control rooms to control traffic.
- It provides efficient traffic management by optimizing traffic flow.
- Reduces traffic congestions, improves overall road safety and reduces risk of accidents.
- IoT devices and sensors provide real-time information about road conditions to drivers and traffic control room for making better decisions.
- Informs drivers about congested areas and road hazards.
- Radio Frequency Identification (RFID) tags placed on the cars gain access to gated entrances. The purpose of using RFID tags is to save time through automation at gated entrances.
- Surveillance cameras can detect and capture images of traffic rules violators so that traffic authorities can take proper action against them.

5.3.4 Application of IoT in Healthcare Industry

IoT technology combined with AI are revolutionizing the healthcare industry all over the world. Advancement in healthcare has given birth to a new concept called Internet of Medical Things (IoMT). Healthcare is extremely important as heart problems, cancer and many other diseases are increasing in Pakistan, especially in big cities. There are many hospitals in Pakistan that are already equipped with IoT devices. We need to implement the IoT-based healthcare system in all the hospitals of our country to provide better healthcare service to the community. IoT improves productivity of doctors and medical professionals and provides efficient hospital workflow as well. It has many applications in healthcare industry.

The following are the applications and benefits of IoT in healthcare industry.

- IoT combined with AI is transforming the way patients' critical data is collected in healthcare industry. All the data collected is saved in cloud. Here, smart algorithms of machine language (ML) analyze the data for patterns to diagnose the disease, recommend medication or additional tests.
- IoT devices can track patients' blood pressure, heart beat rate, oxygen level, etc. in real-time. This information is used by medical professionals to monitor various health issues for proper treatment.
- Wearables equipped with sensors can be used for remote patient monitoring to provide medical care while patient is at home. Real-time patient tracking reduces unnecessary hospital visits and saves time and money.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

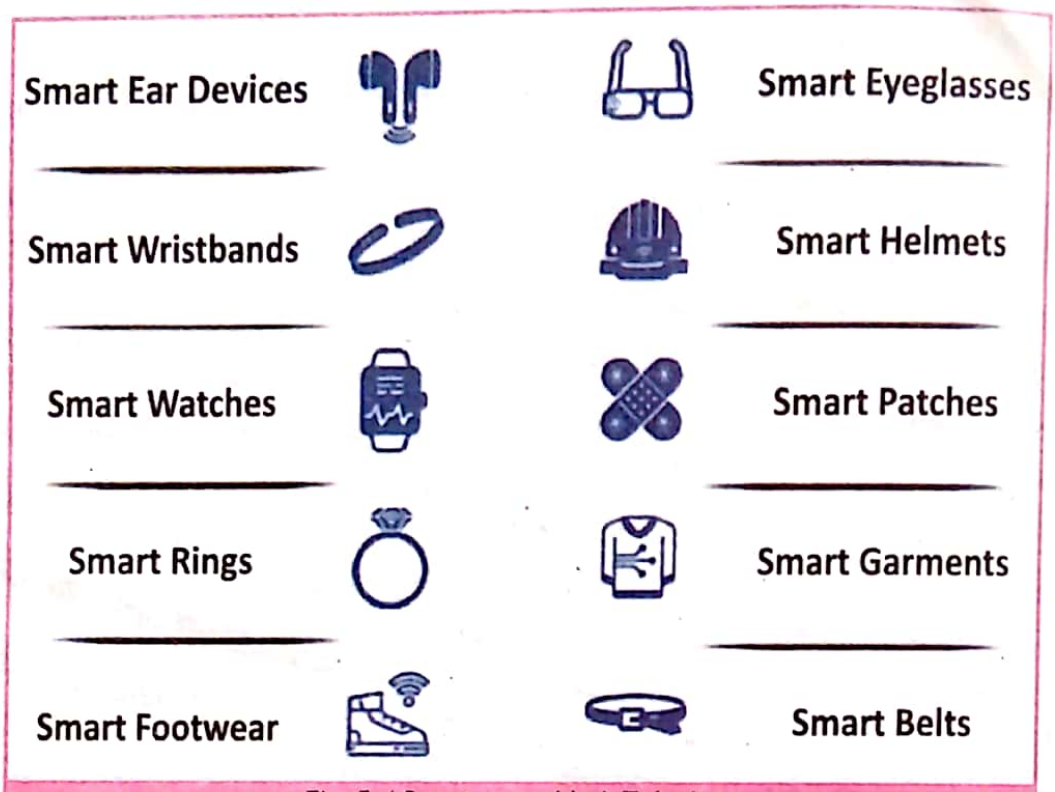


Fig. 5.4 Smart wearable IoT devices

- Modern hospitals are equipped with smart beds with various types of sensors to observe vital signs of patients in real-time. These sensors alert the nurse or doctor to urgently attend the patient if needed.

Fig. 5.5 Smart medical bed equipped with various sensors

- One of the recent developments in healthcare industry is introduction of smart ambulances fitted with various sensors. These ambulances provide diagnosis and treatment during transport to hospital in emergencies. In the past many patients have perished due to lack of required support system during transport. Ambulances fitted with sensors send critical data of patient to healthcare centers so that patient can receive proper treatment while remaining in the ambulance.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

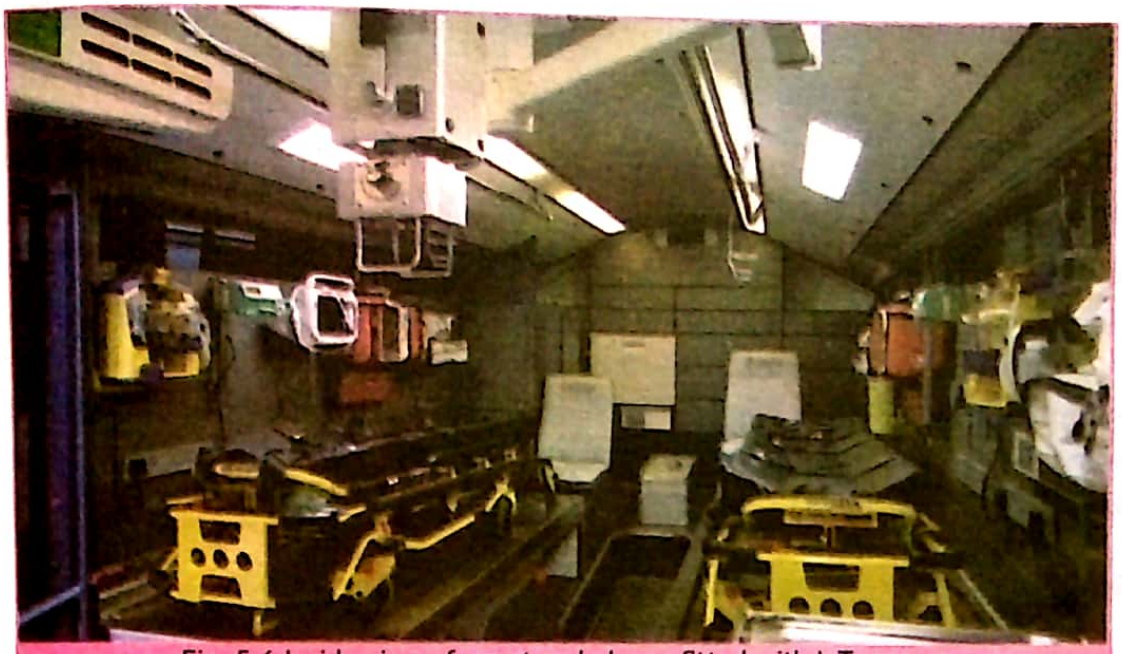


Fig. 5.6 Inside view of smart ambulance fitted with IoT sensors

5.3.5 Application of IoT in Manufacturing Industry

The future of IoT combined with AI is very promising in manufacturing industry. Today, big transformation in going on in manufacturing sector with the integration of IoT technology. It is essential to use IoT in manufacturing in Pakistan as well to improve efficiency, productivity and safety of workers and reduce production cost.

Fig. 5.7 Smart manufacturing factory

The following are the applications and benefits of IoT in manufacturing.

- It enables automation and optimization of processes involved in manufacturing.
- Large variety of sensors are used in a network to enable wireless communication between each other, collect and share data and efficent coordination of production activities.
- Robots are used to enhance accuracy and speed up manufacturing processes.
- It streamlines workflow in factories with minimal human intervention.
- Improves product quality using robotic systems, sensors, cameras and AI-based machine learning.
- Improves customer satisfaction by reducing defective products.
- Reduces workplace accidents and injuries caused by manual labor and risky tasks.
- Reduces product cost through automation, operational efficiency and reduction in labor cost.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- IoT combined with Blockchain technology enables efficient supply chain of products.
- Use of IoT devices in machines and other manufacturing equipment can reduce energy consumption which results in product cost reduction.

5.3.6 Application of IoT in Homes

Continuous developments and innovations in IoT has enabled the development of smart homes. Smart homes are residences equipped with IoT enabled home appliances connected to a central hub. The central hub receives data from sensors and processes it to take decisions. The appliances can communicate wirelessly with each other and the homeowners. IoT enabled devices in a smart home include TVs, ovens, washing machines, refrigerators, lights, fans, air-conditioning systems, security cameras, etc. In Pakistan some IT companies are providing services to transform traditional homes into smart homes.

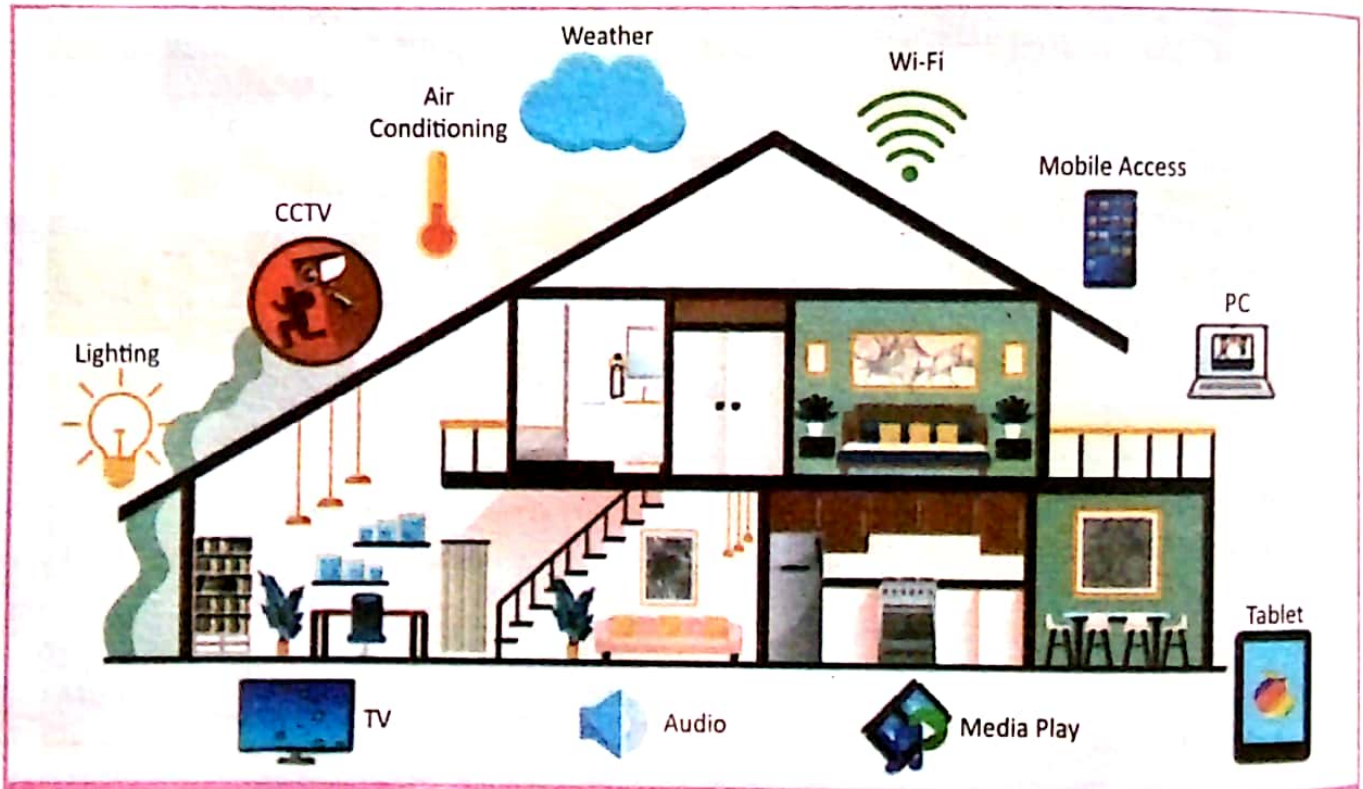


Fig. 5.8 Smart home equipped with various sensors

The following are the benefits of using IoT in smart homes.

- IoT provides convenience and efficiency in operation and control of home devices through smartphone or web interfaces providing better living environment.
- Homeowners can remotely control and monitor the devices when they are away.
- It reduces energy consumption resulting in reduced electricity bills. For example, sensors can detect whether a person is in the room or not and automatically turn on or turn off the air-conditioning or lights.
- Smart cameras, smart door locks, motions sensors and videos can alert the homeowner in real-time for security and safety.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.3.7 Application of IoT to Control Air Pollution

Air pollution is one of the biggest problem in developed and developing countries. It is also a serious issue particularly in the metropolitan cities of Pakistan as health problems related with air pollution are growing at a fast rate. Air pollution is caused by dangerous gases emitted from industries, vehicles and homes. It causes heart diseases, lung cancer and eye irritation. It can also cause allergic reactions near the throat and eye.

IoT-based air pollution monitoring system can provide a solution to health problems related with air pollution. It collects data from various sensors in real-time and processes it for air pollution forecasts to see the trends and control it. IoT technology helps in taking intelligent and quick response for health and safety of community.

Fig. 5.9 IoT based air pollution monitoring system

5.3.8 Application of IoT in Natural Disaster Management

Natural disasters occur in many places and many people loss life. IoT can help in anticipating disasters and provide important information to emergency responders to effectively manage them. IoT devices play an important role in disaster management and planning to minimize the impact and protect infrastructure and loss of human life. Natural disasters include earthquakes, thunderstorm, floods and landslides due to heavy rains which are very common in Pakistan.

Fig. 5.10 Robot in earth quake rescue operation

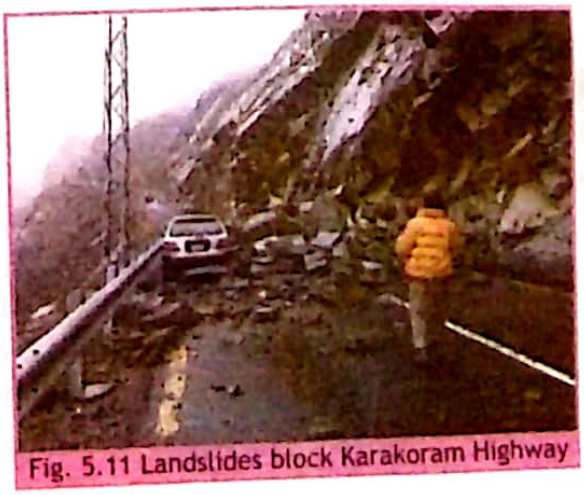


Fig. 5.11 Landslides block Karakoram Highway

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

The following need to be implemented to manage disasters.

Real-time data collection and processing

IoT devices can collect environmental data that can be analyzed and processed to provide critical information. This information can help the concerned emergency authorities to take appropriate decisions according to the environmental conditions.

Monitoring infrastructure

The IoT devices embedded with sensors monitor infrastructure such as dams, bridges and buildings. The data collected from environmental conditions is used in decision-making to protect the infrastructure and save human life.

Search and rescue

Fig. 5.12 Pak Army in flood rescue operation

Fig. 5.13 Drone used for disaster rescue

IoT technology combined with rescue robots and drones equipped with cameras can help emergency responders take quick and effective actions in search and recue of human beings after disasters.

Pakistan National Disaster Management Authority (NDMA)

Pakistan NDMA formulates and enforces national disaster policies at federal and provincial levels. It collaborates with various government ministries, Pak Army and United Nations-based organizations to jointly coordinate efforts to conduct its disaster management, search and rescue and wide range of humanitarian operations in Pakistan and abroad.

Fig. 5.14 Pakistan NDMA Logo

Pakistan NDMA Disaster Alert Mobile App

Pakistan recently introduced the NDMA mobile application, designed to provide essential early warnings and advisories directly to users' devices. Developed by NDMA, this app aims to empower individuals and communities with vital information to effectively prepare for

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

and respond to potential disasters. The NDMA mobile app allows you to stay informed and safe at your fingertips.

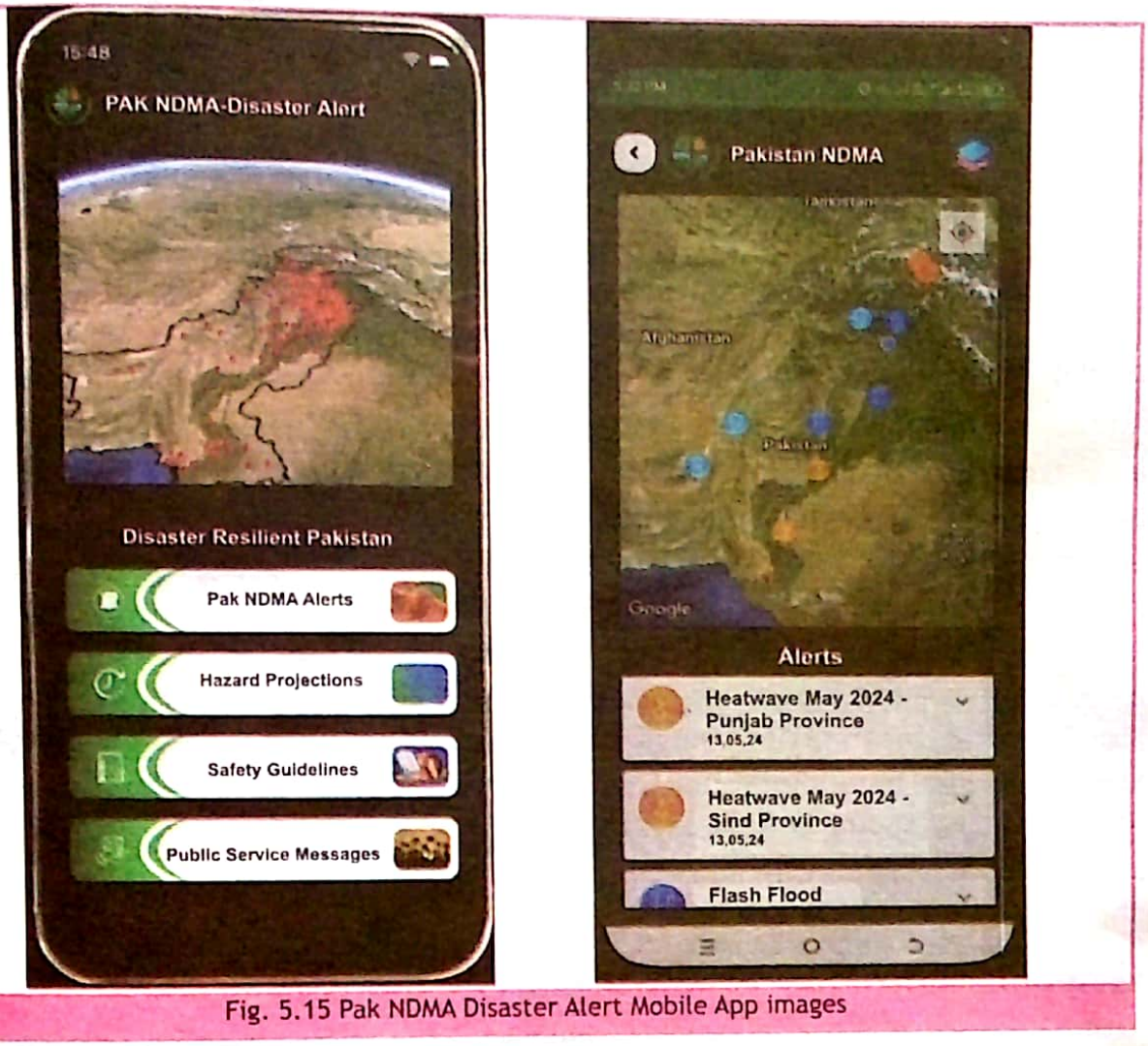


Fig. 5.15 Pak NDMA Disaster Alert Mobile App images

5.4 Blockchain

A Blockchain application is a decentralized application developed on Blockchain technology. It is an innovative approach that relies on distributed ledger (database) of records or transactions that is shared among many network users. The ledger continuously grows as more records known as blocks are added. Blocks are chained together using cryptography to enhance security against data fraud and increase transparency. The use of distributed ledger ensures that once data is recorded it cannot be changed. This reduces vulnerability to cyber attacks. Blockchain provides reliable solutions across diverse industries.

5.5 Blockchain Applications Applicable to Pakistan

The following are some of the areas where Blockchain technology can be implemented in Pakistan.

5.5.1 Application of Blockchain in Voting

Pakistan has faced many issues related with traditional paper ballot system in the past. These

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

include allegation of rigging and manipulations. The use of Blockchain technology can transform voting system by ensuring security and integrity. Blockchain technology can help in voting by ensuring accuracy of voting process. The transparency and immutability of Blockchain can prevent any fraudulent activities. Voters can have confidence that their vote is counted and that the results are accurate. Blockchain can make the voting process more efficient and reduce cost as well.

Fig. 5.16 Blockchain in Voting

5.5.2 Application of Blockchain in Healthcare

Most of the hospitals in Pakistan and worldwide face data security issues in healthcare services. Blockchain technology can revolutionize the way healthcare data is stored. Integration of Blockchain technology with healthcare services can provide increased privacy, transparency and accessibility for patient data. Sensitive medical data, diagnostic images, patient records and all the confidential information must be protected. Blockchain ensures data protection by cryptographic algorithms and the ability of immutability reducing the risk of data loss associated with unauthorized access and breach.

Fig. 5.17 Blockchain in Healthcare

5.5.3 Application of Blockchain in Banking Industry

The digital transformation of banking sector has provided smooth transactions and personalized services. Online banking has simplified daily transfers and transactions reducing frequent trips to banks or ATMs. This digital revolution in banking sector has many benefits but at the same time it has exposed banks to risks of data breaches and cyberthreats. These problems are faced all over the world in banking sector and in Pakistan as well. Blockchain technology integrated with cyber security can provide the solution to these problems. The innovative approach of Blockchain technology can make operation of banks more secure, transparent, efficient and cost effective and eliminate financial frauds.

Fig. 5.18 Blockchain in Banking

5.5.4 Application of Blockchain in Education

Blockchain technology having the capability of distributed ledger, enhanced security and transparency of records can be effectively utilized in Pakistan to solve various problems in education sector.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

The manual system of keeping record of degrees, diplomas and certificates makes verification very time-consuming and difficult and can lead to use of fake credentials. The use of distributed ledger can make record-keeping more efficient, secure, transparent and also help in eliminating the use of fake credentials. It can make the verification process simple and easy as well.

Fig. 5.19 Blockchain in Education

Blockchain technology can also enhance the security and transparency of students' academic records, administrative process and accessibility of records. The adoption of Blockchain technology will allow students to have complete control over their academic records and certificates and degrees as it is saved on Blockchain instead of servers of educational institutions.

5.6 Cloud Computing

Cloud computing refers to the use of hosted services such as servers, data storage, databases, networking and software over the Internet. These services are maintained by cloud service provider and are available on demand. It allows users to save data, files, images, videos and software on the cloud and access these from anywhere with a laptop or desktop computer, mobile phone or tablet connected to Internet.

The following are some areas where cloud computing technology can be used in Pakistan to stay ahead in today's digital world.

5.6.1 Government and Public Sector

Cloud computing offers effective and secure platform to government and public sector to improve service delivery. Cloud-based solutions provide easy storage, access and analysis of vast amount of data. This helps government and public agencies to make more informed and coordinated decisions. It improves citizens interaction with the government by providing online services to conveniently and efficiently engage with the public.

Fig. 5.20 Cloud Computing in Government Public Service

5.6.2 Banking Industry

Cloud-based solutions minimize the need for expensive on-premise IT infrastructure in banking industry leading to significant cost savings. It has revolutionized the way banks operate, manage data and interact with customers. It enables banks to provide more personalized and efficient online services. It also helps in preventing frauds and cyberattacks.

Fig. 5.21 Cloud Computing in Banking

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.6.3. Education Sector

Cloud computing improves overall institutional efficiency. Cloud-based solutions streamline administrative tasks such as student enrolment, grade management and attendance tracking. The online platform of LMS (Learning Management System) provides easy access to course material, assignments, assessments and announcements and efficient interaction with teachers and administrative staff. It enables online classes, workshops and training, saving costs.

Fig. 5.22 Cloud Computing in Education

5.6.4 Businesses

Cloud computing has revolutionized businesses. It enables businesses to store their data and applications on the cloud for easy access from any location. Mobile access to data via smartphones and other devices allows staff to keep instantly up-to-date with customers and co-workers. It reduces operational costs as businesses do not have to own, manage and maintain on-premise servers. It also provides more cost-effective cyber security for protecting data and applications stored on the cloud.

Fig. 5.23 Cloud Computing in Business

5.6.5 Transport

Cloud-based solutions facilitate intelligent transport systems that streamline operations, improve collaboration and enable adoption to new technologies. Cloud computing applications related with transportation improve fleet management, route planning and freight tracking. E-ticketing enhances passengers' travelling experience. It allows easy access to data and applications with mobile devices having Internet connection which enhances collaboration with partners and customers. Cloud computing makes transportation more efficient, faster, easier and reliable.

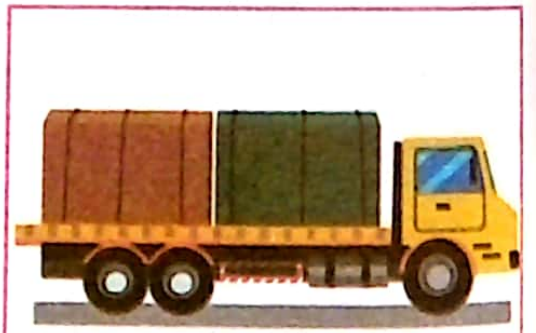


Fig. 5.24 Cloud Computing in Transport

5.6.6 Healthcare

Cloud computing has transformed the healthcare sector with valuable innovations. Cloud computing in hospitals makes it easy for doctors to view and share patients' medical records and update it in real-time. It helps doctors in providing better treatment based on comprehensive past history of patients. Telemedicine platforms enable virtual consultation with patients using

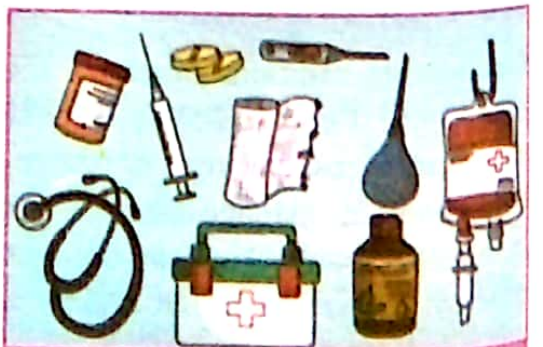


Fig. 5.25 Cloud Computing in Healthcare

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

video conferencing with healthcare providers remotely. It provides healthcare to patients unable to travel to hospitals. It allows patients to access their medical reports, lab testing reports and doctor's notes in real-time.

5.6.7 Media and Entertainment

Media and Entertainment companies are moving to the cloud technology as it provides them platform to deliver digital contents to their customers in an efficient and effective way. Cloud provides fast access to movies, songs and other digital contents from any location with an Internet connection. It allows Entertainment companies to deliver things on time and give users what they want. Media and Entertainment companies are using cloud platforms such as Netflix, YouTube and Spotify to adopt cloud technology and fulfill their customers' demands.

Fig.5.26 Cloud Computing in Media Entertainment

5.6.8 Software Development

Cloud computing provides on-demand computer resources such as storage and processing power to software developers to develop software. It provides significant cost savings for software development as it works on pay-as-you-go basis. Developers can add extra resources whey they are needed or remove unused resources to save costs. It allows software developers to collaborate in real-time from any location using cloud-based tools such as Dropbox and Google Drive for storage and file sharing. It speeds up the work involved in the development of software.

Fig. 5.27 Cloud-based Software Development

5.7 Neural Networks and Deep Learning

Neural networks and deep learning have some differences but they are closely related. Both of them are subset of Machine Learning (ML).

Machine learning is a branch of artificial intelligence that teaches computers to learn from data inputs and experience like humans without direct programming. Machine learning algorithms enable computers to train on data inputs and analyze it to produce accurate output. It recognizes patterns in large databases to build models to automate decision-making.

5.7.1 Neural Networks

Neural networks are simulations of working of human brain. They are based on the functioning of neurons in the human brain. It is a network of neurons that receive and send information throughout the network to make predictions and solve complex problems in a humanlike manner.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

A neuron can be defined as a computational unit of a neural network.

Neural networks have layered approach to process information and produce output. Neural network comprises of the following three layers of neurons or nodes.

- **Input layer:** Input layer consists of one or more neurons. Each neuron receives raw data from the outside world. The raw data can be text, image, video, etc. The input data is passed on to the hidden layer. Each neuron of input layer represents a feature of input data.
- **Hidden layers:** There are one or multiple hidden layers which are between the input and output layers. Number of hidden layers and the number of neurons in each layer depend on the complexity of problem.
- **Output layer:** This is the final layer that produces the output. The output layer can have a single neuron or multiple neurons depending on the task. For example, for classifying whether an email is spam or not, the output layer will have only one neuron. Problems that require multiple output will have more than one neuron in the output layer.

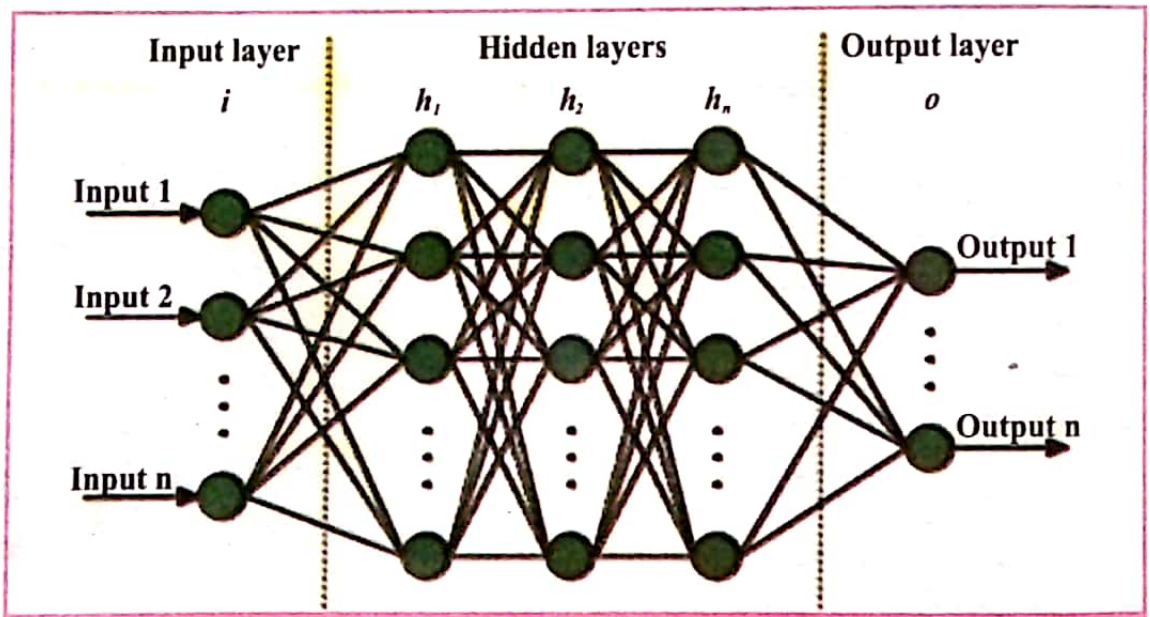


Fig. 5.28 Neural Network

Neurons of input layer receive input and pass it to the hidden layer through the synapse connection. Synapse is a small gap between the layers of neural network that allows information to pass from one layer to the next.

Weights

Weights represent numerical values associated with the connections between neurons. Weights determine the influence that one neuron's output has on another neuron's input in the next layer. Weights act as coefficients that can increase or decrease the importance of specific information. Weights are adjusted to minimize the difference between network's prediction and the actual output. The purpose of using weights is to fine-tune the neural network to make accurate predictions.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Biases

Biases are numerical constants associated with each neuron. Biases are added to neuron's output. The purpose of biases is to provide adaptability to ensure that the network can learn and make better predictions. Suppose a neuron processes the brightness of an image pixel. It will only activate when the pixel's brightness reaches a certain threshold. The bias can allow the neuron to activate even if brightness is below the threshold. Biases provide flexibility to neurons to activate in various input conditions. This helps in handling complex patterns in input data.

Activation Function

When a neuron receives input values, the activation function determines whether the neuron should be activated or not. In other words, the function decides if the input to a neuron is important in predicting the output. The activation function applies mathematical operations to weights associated with the inputs and outputs a value. If the output value of function reaches a threshold, the neuron is activated and the output value is passed on to the next hidden layer or used as final output. Otherwise, the neuron is deactivated and its value is not passed on. Activation functions are used in mapping the input values to a desired output in the last layer.

Loss Function.

Loss function determines how good a neural network model is in solving a problem. It means how well a neural network's predicted outputs match with the true outputs during the machine learning process. It evaluates the performance of a neural network model and helps in optimizing it by minimizing the loss. Minimizing the loss means reducing the mistakes on the learning data.

How a Neural Network is Trained?

Neural network models represent AI algorithms that accept a set of input data and predict output rather than directly implementing algorithms in programming languages. The neural networks learn through two processes known as feedforwad propagation and backpropagation.

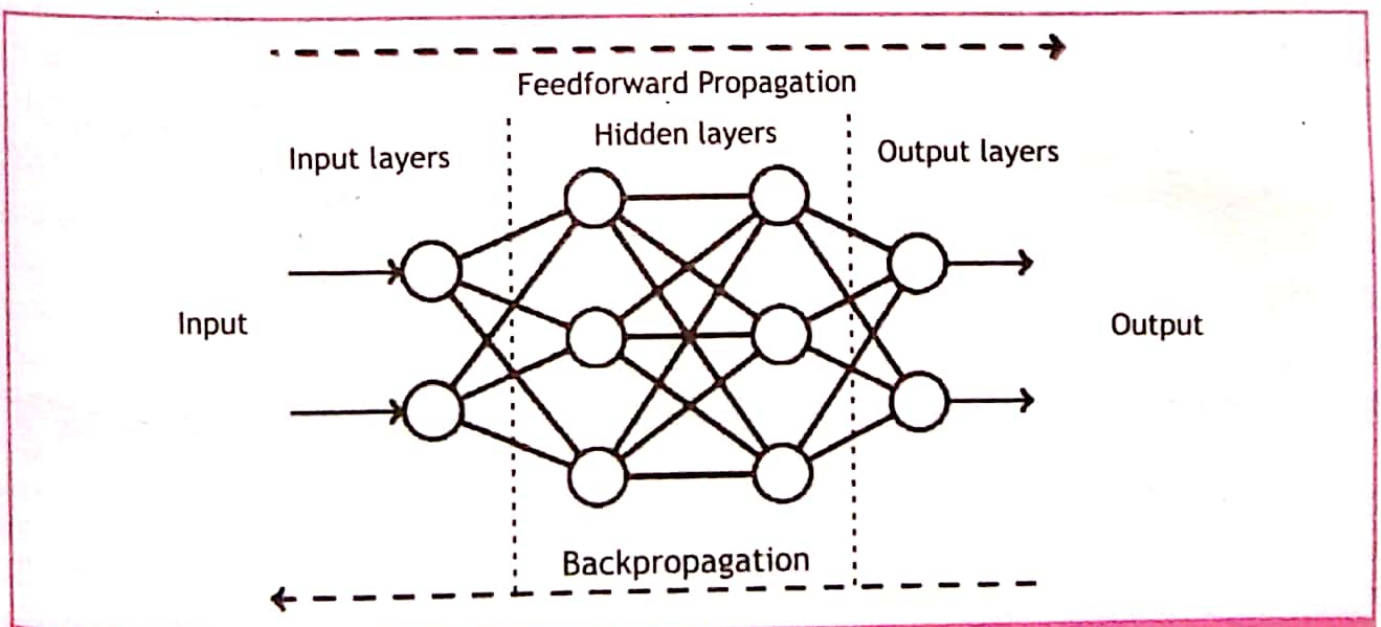


Fig. 5.29 Feedforward and Backpropagation Propagation

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Feedforward Propagation

In feedforward propagation information is fed in the forward direction. It is passed on from the input data layer and then through the hidden layers and finally to the output layer. Neurons in each layer accept input data and process it using activation function. Based on the output values of activation function, it is decided whether the output values should be passed on to the neurons in the next layer or not. Those neurons whose output values are more than a certain threshold pass on the output value to the neurons in the next layer. Each neuron of a hidden layer receives input from the previous layer, processes it in the same way and passes on to the next layer till the final result is produced by the output layer.

Backpropagation

Backpropagation involves comparing the output produced by the output layer with the actual output. Loss function determines how well the network model is performing by matching the predicted output with actual output the network is meant to produce. The loss function is used to optimize the performance by minimizing the loss. The weights of the neurons are modified working from the output layer through the hidden layers and finally the input layer. The backpropagation causes the neural network perform better by reducing the difference between the actual output and the predicted output.

After the network has been trained with various sets of input data, it is ready to accept new sets of data it has never seen before and processes it to predict output.

5.7.2 Real-life Example of Application of Neural Network

The real-life example presented here is about evaluating the price of a house. There are many parameters that influence the price of a house. We are going to consider the following four parameters only to keep the example simple.

- Area (Square feet)
- Number of bedrooms
- Distance to city center (Km)
- House age

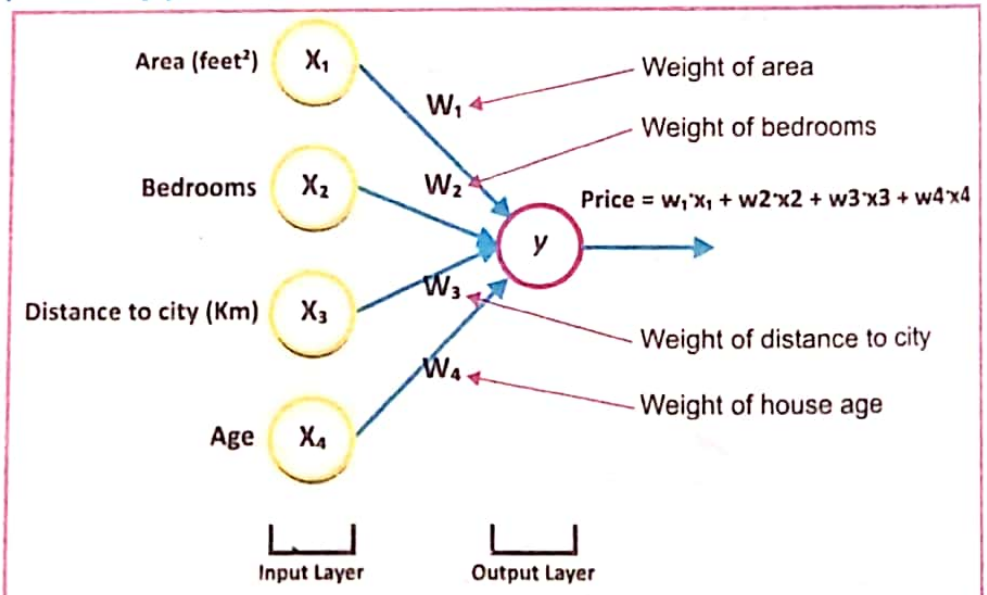


Fig. 5.30 Simple neural network to predict price of a house

A simple weighted sum model of neural network to predict the price of a house is shown in Fig.5.30. This neural network model can be trained to predict accurate result following the steps that have been described earlier.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.7.3 Deep Learning and Neural Networks

Neural networks provide foundational building blocks for machine learning whereas deep learning emphasizes the depth of the neural network. Deep learning models employ neural network architecture. Therefore, they are often called deep neural networks. Without neural networks there will be no deep learning.

5.7.4 Deep Learning (DL)

Deep learning AI technology has revolutionized the world in solving complex problems. It uses deep neural networks to enable computers to autonomously extract hidden patterns and features from huge amount of data sets for making decisions. It provides human-like thinking ability for developing intelligent systems. Compared to machine learning, deep learning requires vast amount of data sets, takes more training time and requires high performance computers with GPU (Graphical Processing Unit). The GPU can process multiple data sets at the same time. Deep learning algorithms improve their decision making capabilities overtime as more sample data is input in the computer.

5.7.5 Applications of Deep Learning

Deep learning applications are used in many industries. Many people are using it in their everyday life. Following are some popular applications of Deep Learning.

1. Natural Language Processing (NLP)

NLP is an important application of DL. It allows machines to understand human language. It can interpret text and speech. DL-based NLP teaches computers and robots to provide suitable responses to linguistic inputs. Some widely used applications of NLP are language translators, sentiment analysis and customer service chatbots.

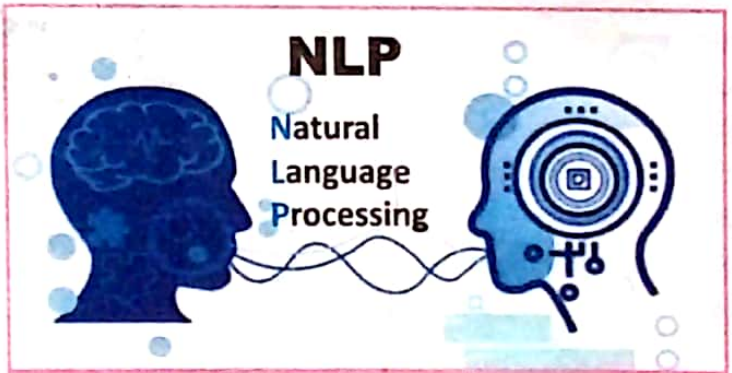


Fig. 5.31 Natural Language Processing

2. Healthcare

DL-based applications are extensively used in healthcare sector by medical professionals to achieve higher quality of service and provide better healthcare to patients. it is used for diagnosis of life-threatening diseases such as cancer and diabetic through medical imaging. It is also used for acute disease detection, smart health monitoring and medical research.

Fig. 5.32 DL applications in healthcare

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3. Fraud Detection

DL can identify fraud and provide protection against it. For example, it can detect suspicious attempts to login into your account and inform you about it. PayPal used DL-based predictive analytics technology to detect and prevent fraudulent activity. DL has the ability to make fraud more predictable and avoidable.

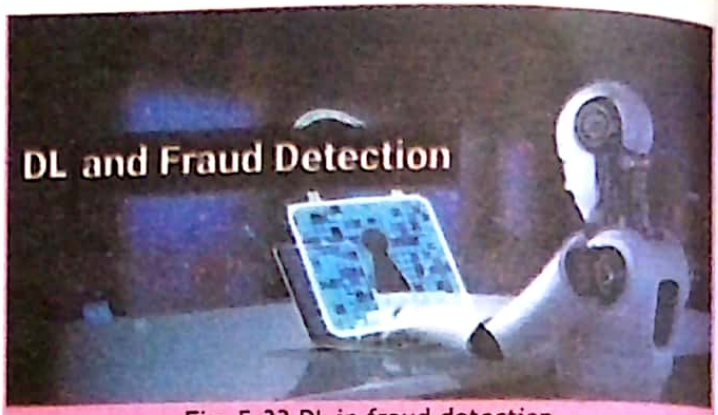


Fig. 5.33 DL in fraud detection

4. Self-driving Car

Self-driving car is an autonomous vehicle based on DL technology that is capable of sensing its environment and operates without human involvement. It uses various sensors to calculate distance between nearby objects. It uses data from cameras, sensors and geo-mapping that enable it to detect traffic lights, recognize signs and avoid pedestrians. Self-driving car can be very helpful to particularly elderly and physically disabled people for travelling. Tesla is an example of self-driving car designed and manufactured by Tesla Inc.

Fig. 5.34 Self-driving Car

5. Robotics

Robotics is a branch of computer science and engineering that involves designing, development, manufacture and operation of a robot. DL technology is heavily used for building robots that can perform specific tasks with little or no human intervention. A robot has various sensors, cameras, microphone and it is controlled by a computer system. Robots are used in manufacturing industry, transporting goods in warehouses,

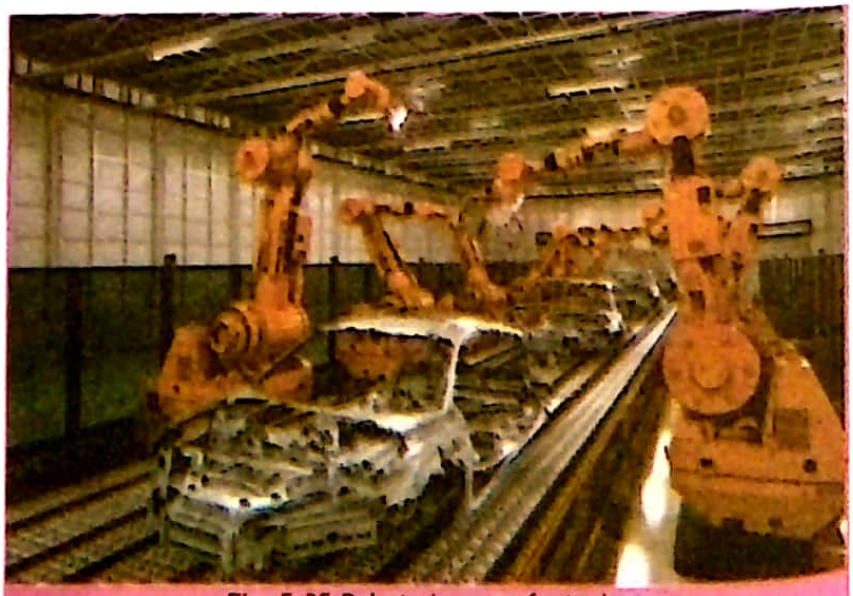


Fig. 5.35 Robots in manufacturing

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

hospitality industry, packaging medicine, performing operations, handling hazardous material, working in extreme temperature, etc.

6. Entertainment

DL applications are used in entertainment platforms for content personalization. DL-based algorithms can analyze vast amount of user data such as browsing history, preference, interests and behavior to provide personalized recommendations. Companies such as Amazon, YouTube, Netflix and Spotify provide personalized movies, songs and videos to enhance their customers' experience.

Fig. 5.36 DL in entertainment

5.7.6 Relationship Between AI, ML, Neural Network and DL

Although the relationship between AI, ML, Neural Network and DL are distinct but they form a close relationship.

- AI is at the core of the relationship used to create machines that can simulate human intelligence. There are many applications of AI. Some of these are Robotics, Computer Vision and Virtual Assistants.
- ML is a subset of AI that focus on the development of algorithms to learn from training data without explicit programming. Some examples of ML are Email Spam Filtering, Online Fraud Detection and Product Recommendation.
- Neural network is a subset of ML. Neural networks simulate human brain to develop algorithms that recognize patterns to help in making intelligent decisions. Examples of application of neural networks are image recognition, Stock Market Prediction and Weather Forecasting.

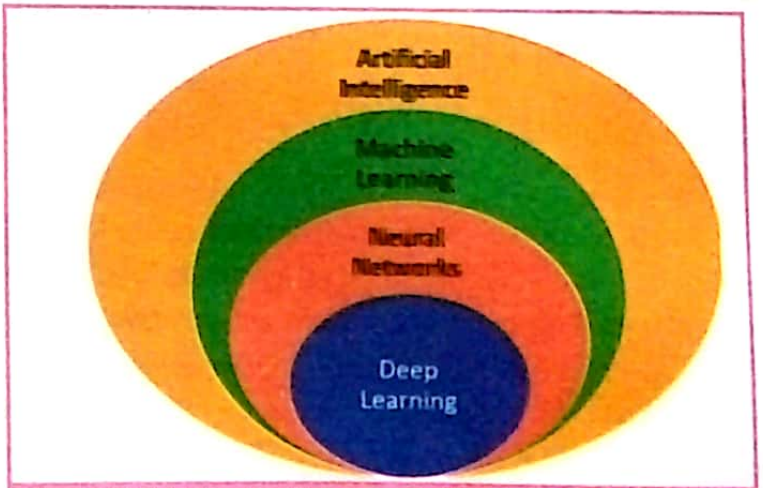


Fig. 5.37 Relationship Between AI, ML, Neural Network, DL

- Deep Learning is a specialized from of ML that uses neural networks with multiple hidden layers. It is particularly used in handling large datasets. Some of its applications are Language Translation, Natural Language Processing, Self-driving Car, Healthcare Diagnostics and playing strategic games such as Chess.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.8 Data Sharing and Privacy

5.8.1 Data sharing

Data sharing means making the same data available to multiple users, applications and organizations. Data sharing has many benefits.

Benefits of data sharing

Fig. 5.38 Data sharing

- Data sharing improves efficiency and collaboration within an organization, with business partners and with wider community. It promotes transparency, research and enables businesses to offer better services to the society. For example sharing data in health sector will be helpful in medical research to provide better healthcare services. Sharing result data by institutions will reveal their academic achievements.
- Data sharing has significant positive influence on decision-making abilities of policymakers, managers and teams. Effective decision-making is essential for successful organizational performance.
- It allows to build relationship with customers. It improves marketing of products and services by reaching out people with right offer.
- Data sharing by government plays an important role in formulating public policies and planning delivery of services. Sharing the right data and collaboration among government agencies results in improving citizens lives.

5.8.2 Data Privacy

Data privacy generally means the ability of a person to control his personal information and get to know when, how and what personal information is shared with others. Data privacy applies to sensitive data such as Personal Identifiable Information (PII), Personal Health Information (PHI) and financial information of businesses and organizations. PII includes information such as names, ID numbers, email or postal addresses, birthdates and contact numbers. Sensitive data about employees, customers and shareholders plays an important role in operations and finances of businesses and organizations.

Fig. 5.39 Data Privacy

5.8.3 Data Sharing and Data Privacy Conflicts

With the increased use of Internet over the year, the importance of data privacy has increased as

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

websites, applications and social media platforms collect and store personal data about users to provide services. Ordering food online or purchasing items from online stores is very common for which people share personal information. Data sharing has many benefits but it comes with many risks and challenges. Organization need to identify the risks when sharing personal data and know who has access to it and how it is used. Shared data can be exposed to potential threats such as unauthorized access, misuse and theft. It is challenging to ensure that data is shared for the right purpose, particularly sensitive and confidential data. Unauthorized and illegal access to shared data can result in invasion of privacy, personal safety risk and it can also damage reputation. Thus, we can say that data sharing and privacy have conflicting requirements.

Fig. 5.40 Data Sharing and Privacy Conflicts

5.8.4 Misuse of Personal Data

Misuse of personal information can have very harmful consequences for individuals, businesses and the whole society. It is of the utmost importance for individuals, organizations and governments to maintain ethical standards and follow data protection regulations when collecting, sharing and processing personal information. Respecting others privacy rights when collecting personal information and ensuring ethical practices contributes to a more trustworthy and safe digital environment.

Fig. 5.41 Misuse of Personal Data

5.8.5 Safeguard Against Information Misuse

Data protection regulations have been put in place by governments all around the world. These include General Data Protection Regulations (GDPR) and the California Consumer Privacy Act (CCPA). GDPR has developed rules and regulations how the personal data of individuals in European Union (EU) can be collected, stored and processed. CCPA regulates that consumers should know what personal data is collected and gives consumers control over their personal data such as right not to disclose or sell it to others. Governments around the world may pass additional data privacy laws in the future as many people think that individuals do not have sufficient control over what happens to their personal data.

Fig. 5.42 Safeguard Against Information Misuse

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

5.8.6 Assessment of Polices to Protect Stakeholders' Interests in Data Privacy Conflicts

The following are the steps to assess data privacy conflicts among stakeholders and find its practical solutions.

1. Identify the Source of Data Privacy Conflicts

It is important to identify the root causes of data privacy conflicts for addressing these appropriately.

The following are the sources that can cause data privacy conflicts.

- Inconsistencies in organization's data governance policies. Data governance refers to how data is collected, stored and processed and who can have access to what kind of data.
- Conflicting standards and expectations related with data quality, security and accessibility among data users and providers. Data quality refers to data accuracy and reliability.
- Ethical and social issues related with collection, use or sharing of personal data.
- Different interpretations of privacy laws, lack of clarity in data handling policies or conflicting interests among stakeholders.
- Misunderstanding how data is being used.
- Different data protection laws and regulations in different countries and regions.

2. Negotiate and Compromise to Resolve Conflicts

The following are the important points to resolve conflicts.

- The stakeholders should negotiate trade-offs for their interests and objectives. They should establish common grounds to seek mutual agreement and compromise.
- Stakeholders should engage in dialog to understand each party's priorities and limitations.
- Stakeholders should balance the need for data access and usability with the need for data protection and encryption.
- They should balance the need for data completeness and accuracy with the need for data minimization and anonymization

Data minimization means to limit the collection to personal information to what is necessary to accomplish a specified purpose.

Data anonymization is the process of altering or removing personally identifiable information from datasets to protect privacy of individuals while enabling its utilization in software testing and analytics. For example, altering or removing the house number from the address of people so that they are not identifiable.

3. Implementing the Solutions

The agreed-upon solutions among the stakeholders should be effectively implemented to resolve data privacy conflicts.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- A data privacy framework should be adopted that implements all the aspects of data governance strategy.
- Data privacy tools such as data encryption, data loss prevention, firewall, etc. should be applied to control information that is being shared.
- Data privacy compliance should be monitored.
- Training programs should be conducted to engage employees at all levels in addressing concerns related with data sharing and gaining their support.
- Feedback should be taken from all the stakeholders for monitoring effectiveness of implementation of solutions and identification of areas for improvement.

5.8.7 Developing Data Governance Policies to Mitigate Data Sharing and Privacy Conflicts

Well-structured data governance policies are essential for handling data in compliance with laws and regulations. Conflicts in policies can cause confusion, inefficiency and even legal issues.

The following are the steps to develop policies for overall management of data collection, availability, usability, integrity and security in an organization.

1. Assess Context

The following points should be considered for assessing context.

- Assess the context in which the current conflicting policies were developed. Determine whether any policies are outdated or were developed for specific scenarios that no longer apply.
- Identify the reasons behind each policy and find out which aspects are open for revision and which are non-negotiable due to legal compliance.
- Determine the best course of action through assessment of the context in which conflicting policies were developed and what were their intended outcomes.
- Identify each policy's objectives and compliance requirements and analyze how they align with the overall goals of the organization.
- Involve the stakeholders to understand their point of view on each policy.
- Access the risks and benefits associated with each policy related with factors such as data privacy and security.

2. Identify the Stakeholders

Identify the stakeholders who will be impacted by the conflicting policies and who will have the authority to modify them. Stakeholders include department heads, IT security teams, legal advisors and any other relevant parties. Engage the stakeholders to get their perspective which is important for understanding the implications of each policy. This will help in building consensus, developing more balanced policies and make certain that all the issues are addressed and aligned with the organization's objectives.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3. Evaluate Risks

it is critical to assess risks associated with each policy. Evaluate the possibility of risks considering security of sensitive information, consequences of non-compliance with the regulations and reputational damage. Consult with stakeholders and legal advisors to get their perspective about risks associated with policies. Document all the decision-making process and clearly communicate to all the stakeholders to maintain transparency and accountability.

4. Propose Solutions

Propose solutions after analyzing the strength and weaknesses of each conflicting policy. Pay close attention to compliance requirements and organizational goals. Solutions may involve suggesting change to one policy, combining elements from two or developing new policy. Ensure that solutions are specific and actionable. Policies that are not clearly stated may lead to misunderstanding and confusion

5. Implement Changes

The final step is the implementation of solutions that have been agreed-upon by the stakeholders. This includes updating documentation and communicating changes in the policies to all the relevant parties. This may also require providing training if needed. It is important to monitor the implementation of changes and obtain feedback to improve data governance within the organizations.

5.8.8 Principles adapted by Organization for Economic Cooperation and Development (OECD) on Data Sharing and Protection of Privacy

The OECD is an international organization of 37 member countries established in Paris in 1961. It was established to develop social and economic policies and international standards for better lives. It focuses on issues such as trade, environment, technology and education. Pakistan is not a member of OECD but has been regularly invited to participate in its meetings.

Fig. 5.43 Organization for Economic Cooperation and Development Logo

The following are the principles adapted by OECD on data sharing and protection of privacy.

- There should be limits to how much personal data can be collected.
- Personal data, when collected, should be accurate and related to the purpose it is being used for.
- The use of personal data should be specified.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- Data should not be used for purposes other than what was specified.
- Data should be kept secure.
- Personal data collection and usage should not be kept secret from individuals.
- Individuals have a number of rights, including the right to know who has their data, to have their data communicated to them, to know why a request for their data is denied, and to have their personal data corrected or erased.
- Anyone who collects data should be held accountable for implementing these principles.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Internet of Things (IoT)** is a network of physical devices having sensors such as smart phones, smart watches, smart security systems, smart home appliances, smart TVs and smart doors, connected through Internet for communication with each other.
- **Smart homes** are residences equipped with IoT enabled home appliances connected to a central hub. The central hub receives data from sensors and processes it to take decisions.
- The **NDMA developed mobile app** aims to empower individuals and communities with vital information to effectively prepare for and respond to potential disasters.
- **Blockchain** is a distributed ledger (database) of records or transactions that is shared among many network users. The ledger continuously grows as more records known as blocks are added. Blocks are chained together using cryptography to enhance security against fraud and increase transparency.
- **Cloud Computing** refers to the use of hosted services such as servers, data storage, databases, networking and software over the Internet. These services are maintained by cloud service providers and are available on demand.
- **Machine Learning** is a branch of artificial intelligence that teaches computers to learn from data inputs and experience like humans without direct programming. Machine learning algorithms enable computers to train on data inputs and analyze it to produce accurate output.
- **Neural Networks** are simulations of working of human brain. They are based on the functioning of neurons in the human brain. It is a network of neurons that receive and send information throughout the network to make predictions and solve complex problems in a humanlike manner.
- **Weights** represent numerical values associated with the connections between neurons. Weights determine the influence that one neuron's output has on another neuron's input in the next layer. Weights act as coefficients that can increase or decrease the importance of specific information.
- **Biases** are numerical constants associated with each neuron. Biases are added to neuron's output. The purpose of biases is to provide adaptability to ensure that the network can learn and make better predictions.
- **Activation Function** applies mathematical operations to weights associated with the inputs and outputs a value. If the output value of function reaches a threshold, the neuron is activated and the output value is passed on to the next hidden layer or used as final output.
- **Loss Function** determines how good a neural network model is in solving a problem. It

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

means how well a neural network's predicted outputs match with the true outputs during the machine learning process. It evaluates the performance of a neural network model and helps in optimizing it by minimizing the loss.

- In **feedforward propagation** information is fed in the forward direction. It is passed on from the input data layer and then through the hidden layers and finally to the output layer.
- **Backpropagation** involves comparing the output produced by the output layer with the actual output. Loss function determines how well the network model is performing by matching the predicted output with actual output the network is meant to produce. The loss function is used to optimize the performance by minimizing the loss.
- **Deep Learning** uses deep neural networks to enable computers to autonomously extract hidden patterns and features from huge amount of data sets for making decisions.
- **Data sharing** means making the same data available to multiple users, applications and organizations.
- **Data privacy** generally means the ability of a person to control his personal information and get to know when, how and what personal information is shared with others.
- **Data Governance** refers to how data is collected, stored and processed and who can have access to what kind of data.
- **Data Minimization** means to limit the collection to personal information to what is necessary to accomplish a specified purpose.
- **Data Anonymization** is the process of altering or removing personally identifiable information from datasets to protect privacy of individuals while enabling its utilization in software testing and analytics.
- The **Organization for Economical Cooperation and Development (OECD)** is an international organization of 37 member countries established in 1961. It was established to develop social and economic policies and international standards for better lives. It focuses on issues such as trade, environment, technology and education.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. Which of the following technologies uses distributed ledger?
   a. Cloud computing
   b. Neural network
   c. IoT
   d. Blockchain
2. Which of the following is used for remote patient monitoring?
   a. Smart medical bed
   b. Smart wearables
   c. Smart ambulance
   d. Drone
3. Which layer feeds the raw data into the neural network?
   a. Hidden layer
   b. Input layer
   c. Network layer
   d. Output layer
4. Which of the following represents numerical values associated with the connections between neurons?
   a. Weight
   b. Biases
   c. Activation function
   d. Loss function
5. Which of the following decides if the input to a neuron is important in predicting the output?
   a. Weight
   b. Biases
   c. Activation function
   d. Loss function
6. Ability of a person to control his personal information is known as:
   a. Data protection
   b. Data privacy
   c. Data copyright
   d. Data transfer
7. Which of the following is used for training robots in performing complex tasks?
   a. Artificial intelligence
   b. Machine learning
   c. Neural network
   d. Deep learning
8. Which of the following alters personally identifiable information in datasets to protect data privacy?
   a. Data minimization
   b. Data anonymization
   c. Machine learning
   d. Deep learning
9. The technology used to enhance security against fraud and increase transparency is:
   a. IoT
   b. Cloud computing
   c. Blockchain
   d. Neural network
10. Which of the following focus on the development of algorithms to learn from training data?
    a. Robotics
    b. Natural language processing
    c. Machine learning
    d. Cloud servers

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Give short answers to the following Short Response Questions (SRQs).

1. Give three reasons how IoT can reshape traditional agriculture and improve farming in Pakistan?
2. Describe three benefits of using IoT in manufacturing industry.
3. Describe how IoT can be helpful in controlling air pollution in Pakistan.
4. Describe the purpose of National Disaster Management Authority (NDMA) mobile app?
5. Differentiate between neural network and deep learning.
6. Briefly describe the applications of deep learning.
7. Describe three benefits of sharing data.
8. Briefly describe how personal data can be misused.
9. What are the benefits of using cloud computing in business sector?
10. Differentiate between data anonymization and data minimization with examples.

Give long answers to the following Extended Response Questions (ERQs).

1. Mention two applications of IoT technology applicable to Pakistan that are not presented in the text book. Why they are needed and what improvement they can bring?
2. Mention two applications of Blockchain technology applicable to Pakistan that are not presented in the text book. Why they are needed and what improvement they can bring?
3. Develop a simple neural network model to predict price of a smart mobile phone based on five specifications such as processor, camera, screen size, battery life, etc.
4. Mention four risks associated with data sharing and privacy in organizations.
5. Briefly explain how data governance policies can be helpful in mitigating data sharing and privacy conflict.
6. Mention five reasons due to which data privacy conflicts can arise among stakeholders in organizations in Pakistan.
7. Which principles should be adapted for data sharing and protection of privacy in Pakistan?

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Activity 1

Case Study Analysis: Provide students with case studies of real-life scenarios that have data sharing and privacy conflicts, such as the Cambridge Analytica scandal, and ask them to evaluate the situation and suggest policy decisions that can help achieve acceptable compromises. This exercise will allow students to practice analyzing and evaluating real-world examples of data sharing and privacy conflicts and develop their critical thinking skills.

Activity 2

Debate: Divide the class into two groups and assign each group a stance on a data sharing and privacy conflict. For example, one group could argue that social media companies should be required to share user data with the government to prevent terrorism, while the other group could argue that this would be a violation of privacy rights. This exercise will allow students to practice arguing their points of view and develop their communication and collaboration skills.

Activity 3

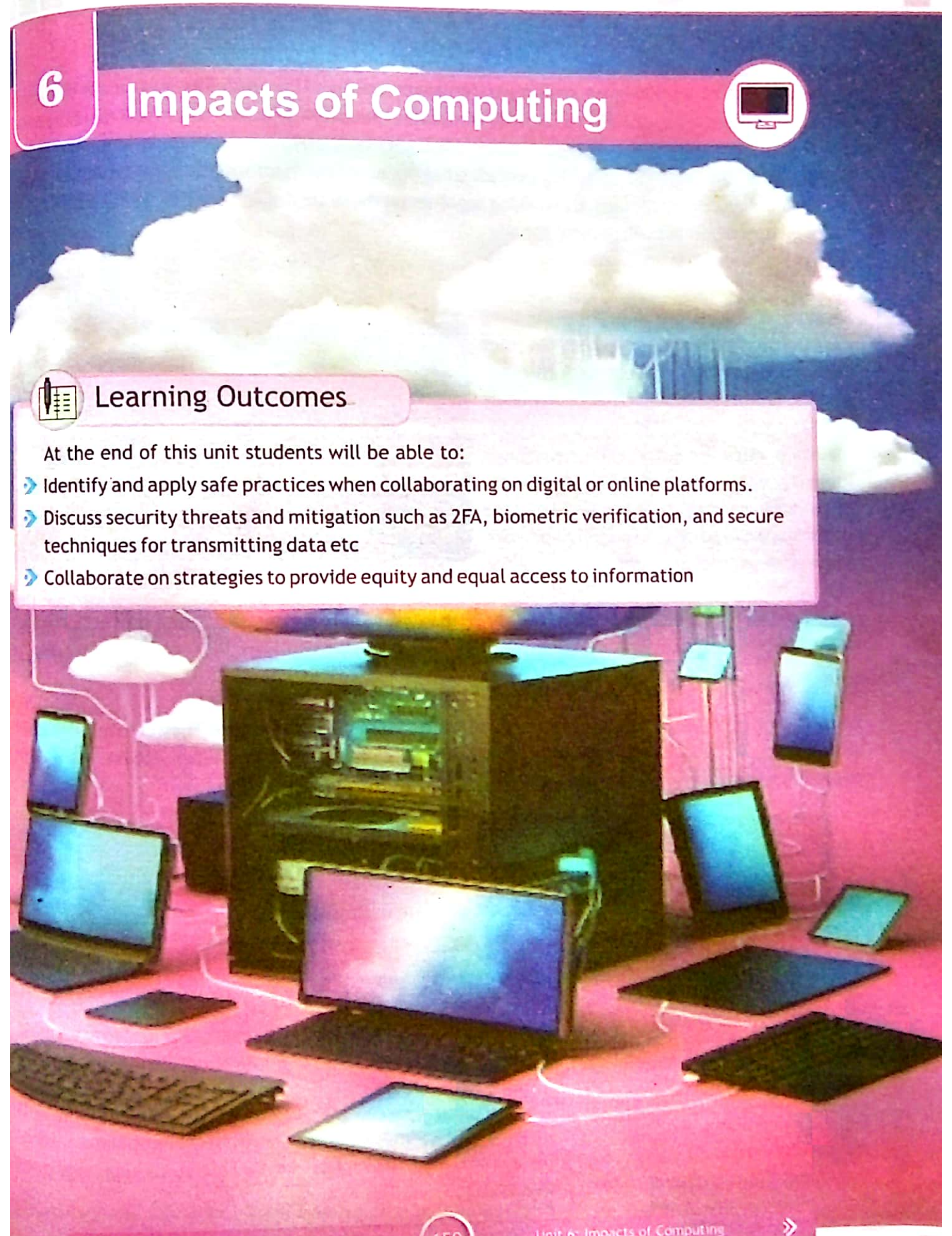
Policy Writing: Ask student to work in groups to draft a policy document that outlines how organizations should handle data sharing and privacy conflicts. The policy document should consider the benefits and drawbacks of data sharing, as well as the importance of protecting individuals' privacy rights. This exercise will allow students to practice working collaboratively, developing policy proposals, and understanding the nuances of data sharing and privacy.

Activity 4

Create an activity that requires taking an example of data sharing and privacy conflicts in the news or online. Students evaluate the situation and suggest policy decisions that could help resolve the conflict.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

6 Impacts of Computing



Learning Outcomes

At the end of this unit students will be able to:

- Identify and apply safe practices when collaborating on digital or online platforms.
- Discuss security threats and mitigation such as 2FA, biometric verification, and secure techniques for transmitting data etc
- Collaborate on strategies to provide equity and equal access to information

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Introduction

In today's digital world, using online platforms for collaboration requires awareness of cyber safety practices. It is crucial to identify and apply security measures to protect participants' privacy. Techniques like two-factor authentication, biometric verification, and secure data transmission are essential. Ensuring everyone has equal access to information, regardless of their background, is also important. By working together on these strategies, we can create a safer and more inclusive online environment for all.

6.1 Security Protocols

Security protocols are vital for protecting data in today's digital world. They set rules and procedures to prevent unauthorized access and tampering. These protocols ensure confidentiality through encryption, maintain data integrity and verify user identity with authentication methods. They also control access, secure data transmission, and ensure compliance with regulations. Additionally, they protect against malware, support data backup and recovery, reduce risks, and build trust. Security protocols are essential for data protection, risk management, and regulatory compliance.

Fig.6.1: Security Protocols

6.1.1 Need of Security Protocols:

Security protocols are crucial for keeping data safe in today's digital world. They create rules and procedures to protect data from unauthorized access, tampering, and interception. Here are some key reasons why security protocols are important for data protection.

Fig.6.2: Need of Security Protocols

Confidentiality:

Security protocols, like encryption, keep data confidential. Encryption changes data into a format only accessible by authorized parties with decryption keys. This stops unauthorized individuals from accessing sensitive information.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Data Integrity:

Data integrity ensures that information remains accurate, consistent, and unaltered during storage, transmission, and processing. Security protocols like encryption protect data integrity by preventing unauthorized changes or fraud.

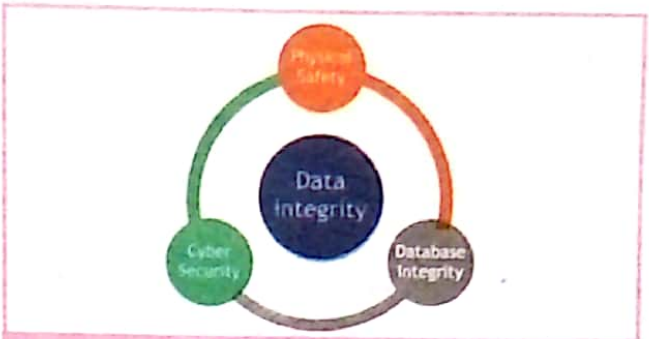


Fig.6.3: Data Integrity

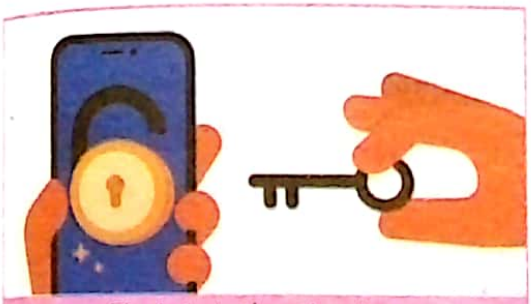


Fig.6.4: Authentication

Authentication:

Security protocols use authentication to verify user or device identities. This ensures that only authorized people or systems access or send data. Common techniques include usernames and passwords, multi-factor authentication, and digital certificates an electronic document used to prove ownership.

Access Control:

Security protocols enable access control mechanisms. These controls decide who can access, change, or send data. It ensures only authorized users access data, systems, and places. It is done through various methods to protect information and privacy, preventing unauthorized access or misuse.

Fig.6.5: Access Control

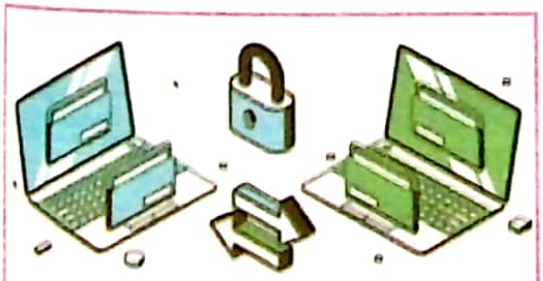


Fig.6.6: Secure Transmission

Secure Transmission:

Sending data online can be risky because hackers might try to spy. However, tools like SSL (Secure Sockets Layer)/TLS (Transport Layer Security) for websites and VPNs (Virtual private networks) for remote connections encrypt your data, making it like a secret code that only you and the intended recipient can understand. This way, even if hackers try to listen, they cannot make sense of the information.

Protection Against Malware and Cyberattacks:

Security protocols help protect against malware, viruses, and cyberattacks. Security protocols are guard against malware and cyberattacks. Firewalls and anti-malware tools are examples of security measures that rely on protocols to function effectively.

Fig.6.7: Protection Against Malware and Cyberattacks

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Data Backup and Recovery:

Data security protocols cover backup and recovery. Regular backups and secure storage rules make sure data can be brought back if it's lost, hardware failure, or there is a cyberattack.

Fig.6.8: Data Backup and Recovery

6.1.2 Information usage and collection:

The main rules for collecting and using data are being clear, getting permission, and following the law. Organizations should explain why they collect data, ask for permission when needed, and protect data well. Breaking these rules can lead to big problems, legally and in how much people trust them.

Fig.6.9: Information usage and collection

Collection:

Collection is the process of gathering information from various sources for analysis and decision-making. It involves methods like surveys, interviews, sensors, or software to collect data in forms such as numbers, text, or images.

Usage:

Data has many uses like offering services, making products better, customizing content, and doing analytics.

- To understand customer preferences and improve products or services.
- To track student performance and enhance learning methods.
- To monitor patient health and develop treatments.
- To train AI models or enhance apps and systems.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

6.2 Risks of Sharing Private Information

Sharing private information can make identity theft more likely. Identity theft happens when someone uses your personal details, like your name or credit card info, without permission for fraud. It can lead to big problems like losing money and harming your reputation. Thieves usually get personal info through various ways.

Phishing:

Scammers may send fraudulent emails, texts, or messages that appear to be from trusted organizations. Scammers asking you to provide personal information like passwords or financial details.

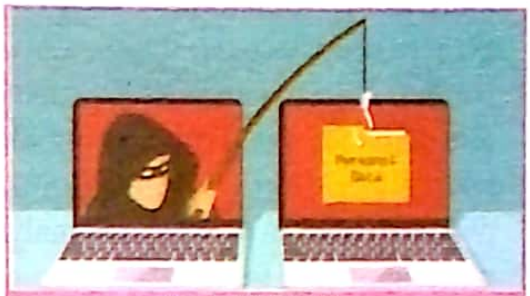


Fig.6.10: Phishing

Fig.6.11: Data Breaches

Data breaches:

When organizations or websites are attacked online, they might lose customer data, and if you're one of their customers, your data could be at risk too. Identity thieves might try to trick people into giving away personal information by pretending to be someone trustworthy.

Lost or stolen items:

Physical items like wallets, laptops, or documents containing personal information can be lost or stolen and then used by identity thieves.

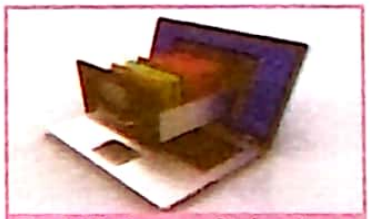


Fig.6.12: Lost or Stolen Items

Fig.6.13: Online Scams

Online scams:

Scammers make fake websites or ads online to fool people into giving personal information or paying money to the wrong places.

6.3 Best Practices to Prevent Identity Theft

Be careful about sharing personal information, especially if asked through email, phone, or social media. Use strong, unique passwords for your online accounts and consider a password manager to keep track of them. Enable multi-factor authentication (MFA) for extra security. Regularly check your financial statements and credit reports for unusual activity. Protect physical items with sensitive information, like IDs, passports, and credit cards. Learn about common scams and be cautious with

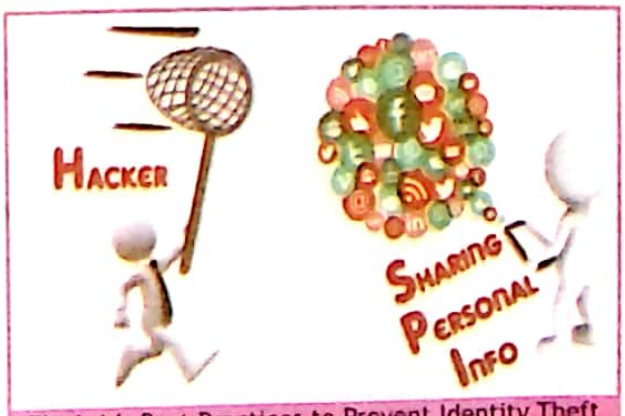


Fig.6.14: Best Practices to Prevent Identity Theft

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

unfamiliar people or organizations. Staying alert helps reduce the risk of identity theft and protects your personal information.

6.4 Cyber-attacks

Cyberattacks put both people and organizations at risk by targeting their security and privacy. They come in different forms like DDoS attacks which overwhelm systems with traffic; ransomware, which encrypts files and demands a ransom; and spyware, which secretly gathers information. Viruses, phishing, and various types of malwares also pose significant threats.

Staying informed and implementing strong cybersecurity measures can help protect against these risks.

Fig.6.15: Cyber-attacks

6.4.1 Types of cyber-attacks:

Certainly, there are various types of cyberattacks that threaten the security and privacy of individuals and organizations.

DDoS (Distributed Denial of Service) Attack:

A DDoS (Distributed Denial of Service) attack is when many infected computers work together to flood a website or system with too much traffic. This overloads the target, making it slow or completely unavailable to regular users.

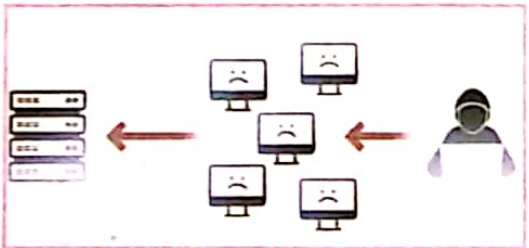


Fig.6.16: DDoS Attack

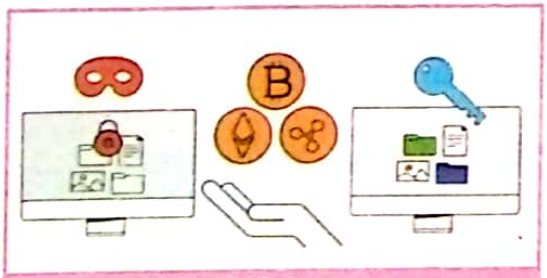


Fig.6.17: Ransomware

Ransomware:

Ransomware is a type of malicious software that locks or encrypts your files so you cannot access them. The attackers then demand money (a ransom) to unlock or decrypt your files.

Spyware:

Spyware is harmful software that secretly collects information about someone without them knowing. It can track what you type, take screenshots, or watch your online activities, and then send this data to a remote attacker.

Fig.6.18: Spyware

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Viruses:

Viruses are programs that copy themselves and infect a computer's files or software. They can spread to other systems when infected files are shared. Viruses can cause various problems, such as corrupting data or crashing the system.

Fig.6.19: Viruses

Fig.6.20: Phishing

Phishing:

Phishing attacks deceive people into giving away sensitive details like passwords or credit card numbers by pretending to be a reliable source. These attacks usually use fake emails, websites, or messages.

Malware (Malicious Software):

Malware refers to harmful software like viruses, spyware, and Trojans, which can damage computer systems or steal information.

Fig.6.21: Viruses

6.5 Security Methods

Two-Factor Authentication (2FA) and Biometric Verification improve security by requiring multiple forms of ID. 2FA needs two separate factors, like a password and a phone app, to confirm identity. If one factor is breached, the other still offers protection. Biometric Verification uses unique physical features, like fingerprints or face scans, for identification. It is convenient and tough to fake but may raise privacy issues. Using both methods together provides strong protection for accessing systems and data.

Fig.6.22: Cyber-attacks

6.5.1 Two-Factor Authentication (2FA):

Two-factor authentication (2FA) enhances security by requiring two separate forms of identification to access a system, account, or app. It usually combines:

1. Something you know, like a password or PIN.
2. Something you have, such as a smartphone, token, or smart card.

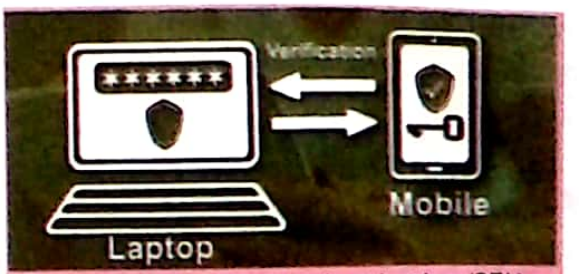


Fig.6.23: Two-Factor Authentication (2FA):

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

3. Something you are, like a fingerprint or facial recognition.

This approach ensures that if one factor (e.g., a password) is compromised, an attacker would still need the second factor to gain access.

6.5.2 Biometric Verification:

Biometric verification uses a person's unique physical or behavioral traits to confirm their identity. This includes fingerprints, facial features, iris patterns, voice, and even typing or walking patterns.

Fig.6.24: Biometric Verification

To access a system or device, users provide a biometric sample, such as a fingerprint or face scan. The system then checks this sample against stored data to see if it matches. Biometric verification is convenient because it eliminates the need for passwords or PINs and is hard to fake. However, it raises privacy concerns due to the sensitive nature of biometric data and how it is stored and used.

6.6 Safe Transmission of Data

Safe data transmission involves securing information as it moves between locations. Cryptography is crucial in this process, converting readable data (plaintext) into an unreadable format (ciphertext) that only authorized users can decode. This ensures the data stays private and unchanged. Common methods include symmetric and asymmetric encryption, and protocols like SSL/TLS, which protect data in transit and prevent unauthorized access or tampering.

6.6.1 Cryptography:

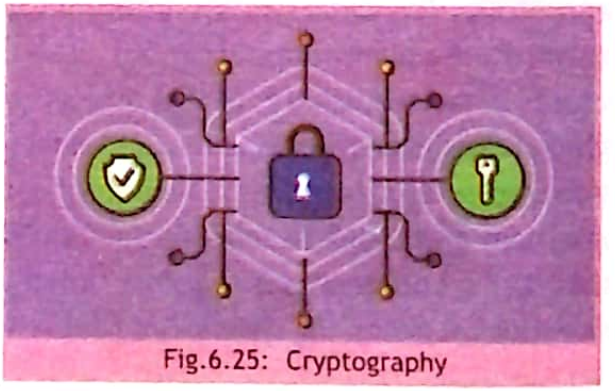


Fig.6.25: Cryptography

Cryptography is the practice of securing communication and data by transforming it into a form that only authorized parties can read. It involves key concepts like plaintext (original message), ciphertext (encrypted message), encryption (converting plaintext to ciphertext), decryption (converting ciphertext back to plaintext), and keys (secret parameters for encryption and decryption). Cryptography can be symmetric, using the same key for encryption and decryption, or asymmetric, using a pair of keys (public and private). Various ciphers, both simple and modern, ensure data integrity and security, making cryptography essential for protecting information.

6.6.2 Types of Cryptography:

Symmetric Cryptography:

Symmetric cryptography uses the same key to encrypt and decrypt data. It is fast and efficient,

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

ideal for large data volumes. The key must be securely shared, as anyone with it can access the encrypted data.

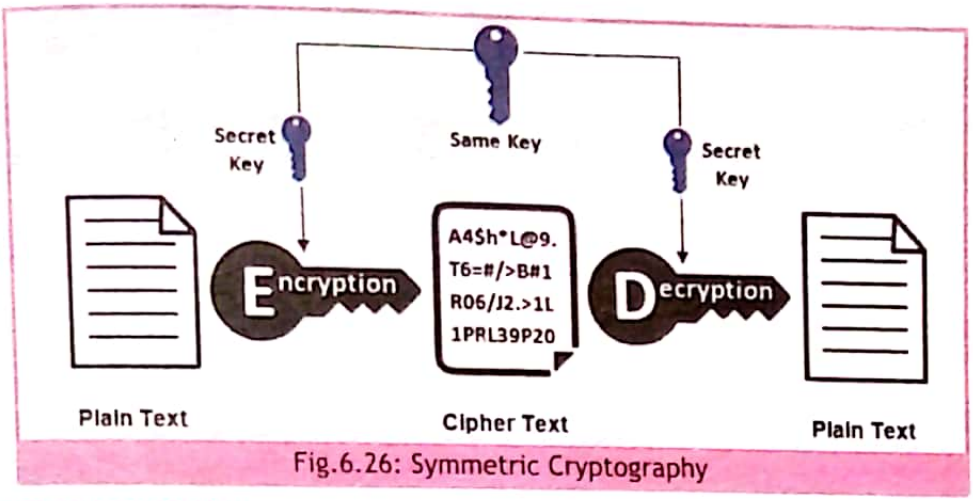


Fig.6.26: Symmetric Cryptography

Asymmetric Cryptography:

Asymmetric cryptography, or public-key cryptography, works with two keys: a public one for encryption and a private one for decryption. This setup enables secure communication without sharing a secret key. While slower than symmetric cryptography, it offers benefits like key distribution and digital signatures for authentication.

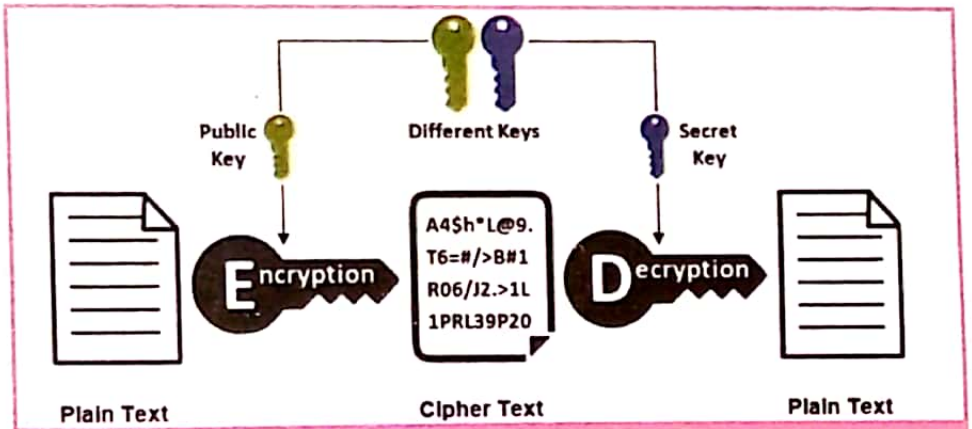


Fig.6.27: Asymmetric Cryptography

6.6.3 Popular Cipher Methods:

Caesar Cipher:

The Caesar Cipher is a simple encryption method where each letter is shifted by a fixed number of positions in the alphabet. For instance, with a shift of 3, 'A' becomes 'D', 'B' becomes 'E', and so forth. It's a basic substitution cipher and among the oldest encryption techniques.

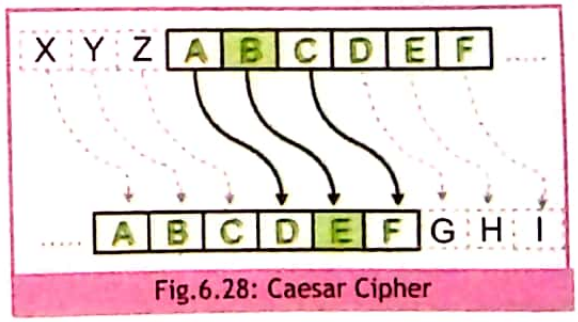


Fig.6.28: Caesar Cipher

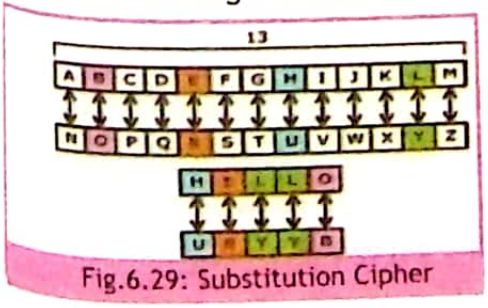


Fig.6.29: Substitution Cipher

Substitution Cipher:

A substitution cipher encrypts by replacing each letter in the plaintext with a different letter. This uses a fixed system called a substitution alphabet. The Caesar Cipher is one type of substitution cipher, but many other substitution alphabets can be used.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Transposition Cipher:

A transposition cipher encrypts by rearranging the positions of characters in the plaintext according to a specific system, without changing the characters themselves. This means the original text is reordered based on a particular pattern.

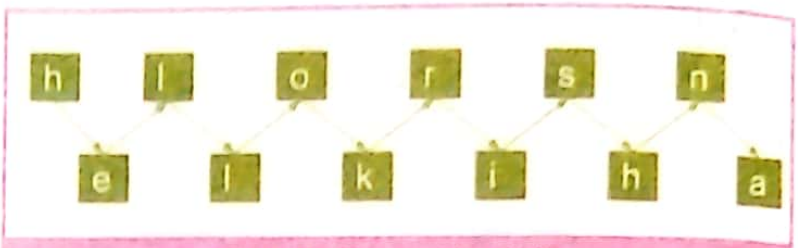


Fig.6.30: Transposition Cipher

Fig.6.31: Vigenère Cipher

Vigenère Cipher:

The Vigenère cipher is a way to hide a message by changing its letters. It uses a keyword to decide how much to change each letter. Each letter in the message is changed based on a matching letter in the keyword, which is repeated as needed.

6.7 Security Protocols

Security protocols are rules and procedures to protect data and communication on networks. They keep sensitive information, like passwords, private and safe from unauthorized access. These protocols also verify user identities and ensure data remains unchanged during transmission. Examples include HTTPS for secure browsing and SSL/TLS for secure data exchange.

6.7.1 Types of protocols:

Transport Layer Security (TLS)/Secure Sockets Layer (SSL):

TLS and its predecessor SSL are cryptographic protocols used to secure internet communication. They offer encryption and authentication to ensure that data between a client and server is confidential and unaltered. TLS is commonly used in web browsers, email clients, and many online services.

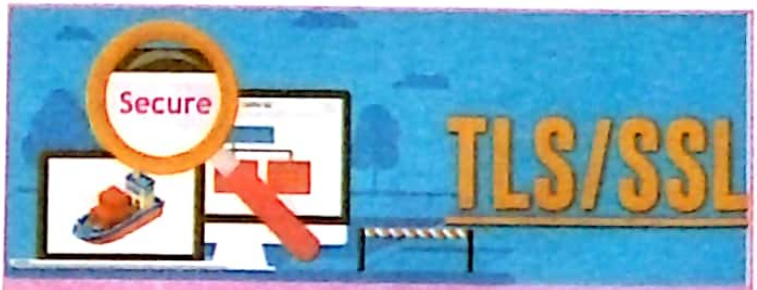


Fig.6.32: TLS/SSL

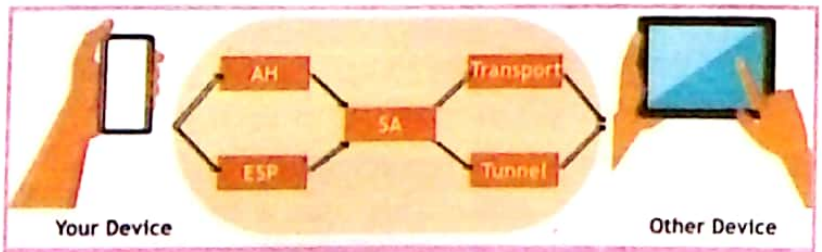


Fig.6.33: Internet Protocol Security

Internet Protocol Security (Ipsec):

IPsec is a set of protocols that secure Internet Protocol (IP) communication. It is often used for VPNs and helps establish secure communication channels between networks, remote clients, and gateways.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

SSH (Secure Shell):

SSH is a protocol for secure remote access to servers and network devices. It offers strong authentication and encrypted communication, making it essential for secure system administration.

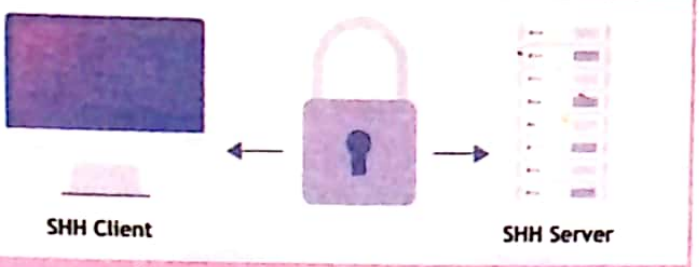


Fig.6.34: SSH (Secure Shell)

Secure File Transfer Protocols:

Secure file transfer protocols safeguard files during transmission across networks. SFTP (SSH File Transfer Protocol) encrypts data for secure and private transfers. FTPS (FTP over SSL) utilizes SSL/TLS encryption to protect files during transfer. Both methods ensure files are transmitted securely, making interception or alteration difficult.

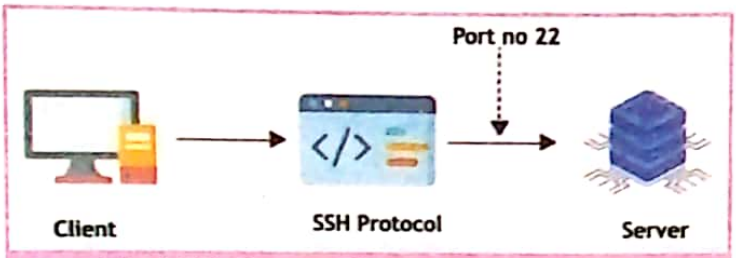


Fig.6.35: Secure File Transfer Protocols

HTTPS (Hypertext Transfer Protocol Secure):

HTTPS (Hypertext Transfer Protocol Secure) is an extension of HTTP used for secure communication over a computer network, typically the internet. It encrypts data exchanged between a web browser and a website, ensuring that sensitive information, such as passwords or payment details, remains confidential and protected from eavesdropping or tampering.

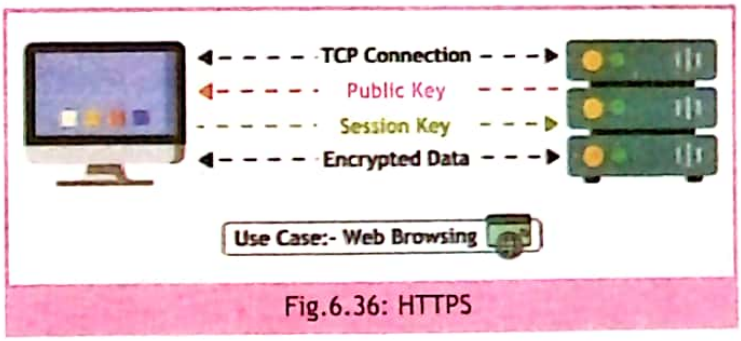


Fig.6.36: HTTPS

6.8 Troubleshoot Security Problems

Troubleshooting security problems means finding and fixing issues to protect systems and apps. It is crucial for safeguarding organizational data and assets. Steps like understanding the setup, checking logs, and running vulnerability tests help address security worries. Following best practices such as keeping software updated, educating users, and planning for incidents boosts overall security and lowers risks. Remember, security is an ongoing process needing constant assessment and adjustment to tackle changing threats. Here are some simple steps and good ways to help you find and fix security issues.

Fig.6.37: Troubleshoot Security Problems

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

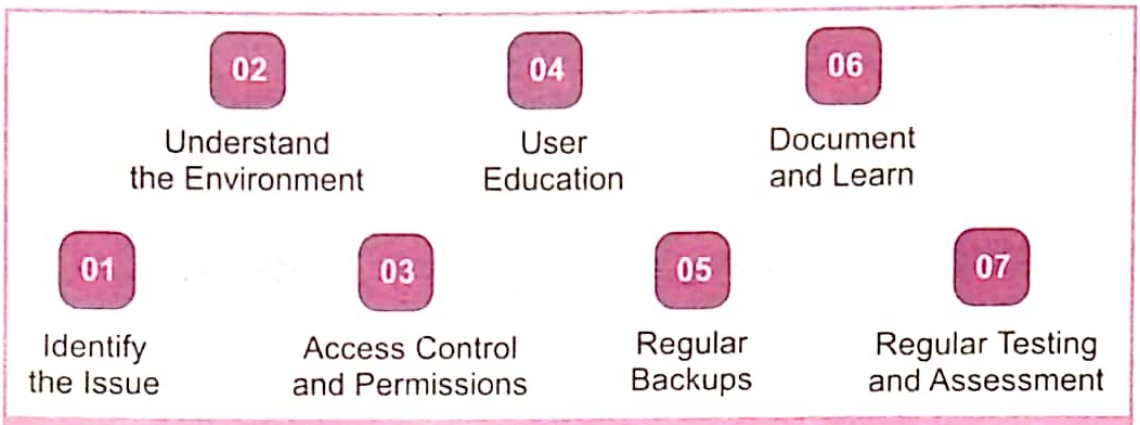


Fig.6.38: Security issues

Identify the Issue:

Start by clearly defining and understanding the security problem. Gather all available information about the symptoms and the systems or applications involved.

Understand the Environment:

Learn about the systems and apps, like how they're set up, what they need to work, and how they're linked in the network. This knowledge will help find weaknesses and ways attackers might try to get in.

Access Control and Permissions:

Check access control lists, user permissions, and roles to ensure that only authorized users have access to sensitive data and functionalities.

User Education:

Teach users and staff about security basics, like not clicking on strange emails or websites, making strong passwords, and reporting security issues quickly.

Regular Backups:

Regularly back up your data and check that you can restore it properly. This is crucial if you lose data because of security problems like ransomware attacks.

Document and Learn:

Write down what happened when you had a problem and what you did to fix it. This written record helps stop similar problems from happening again and makes security better."

Regular Testing and Assessment:

Continuously assess and test your security measures to identify and lighten emerging threats.

6.9 Identifying a Cybersecurity Threat

A common cybersecurity threat is a phishing attack. This involves tricking people into giving away sensitive information like passwords or personal data through fake emails, websites, or messages.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

6.9.1 Various strategies of identifying cybersecurity threats:

Employee Training and Awareness for cybersecurity threats:

Regularly train employees about cybersecurity to help them spot phishing attempts. Encourage them to be cautious of unexpected emails, especially those asking for sensitive information or containing suspicious links or attachments.

Fig.6.39: Employee Training and Awareness

Fig.6.40: Email Filtering and Anti-Phishing Software

Email Filtering and Anti-Phishing Software:

Use strong email filters to catch and block phishing emails. Anti-phishing software can scan emails and spot threats before they reach employees' inboxes.

Secure Communication Protocols:

Use HTTPS for websites and encrypted email services to keep communication secure. Make sure employees check that websites and email senders are genuine before sharing sensitive information.

Fig.6.41: Secure Communication Protocols

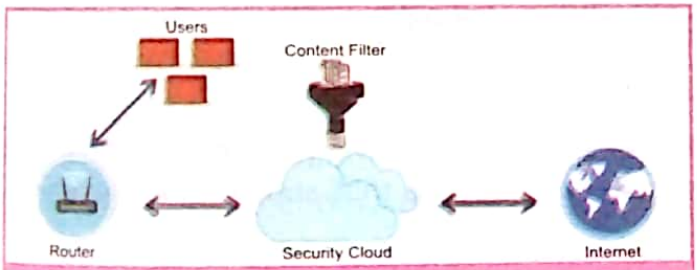


Fig.6.42: Web Content Filtering

Web Content Filtering:

Set up web filters to block access to known phishing sites and harmful content.

Security Updates and Patch Management:

Regularly update all software, operating systems, and applications with the latest security patches. Keeping everything up to date reduces the chances of attacks. These strategies help lower the risk of phishing attacks and improve overall cybersecurity. Stay informed about new threats and adjust your security measures as needed.

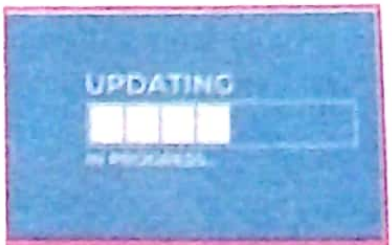


Fig.6.43: Security Updates

6.10 Computational Perspectives

Computational perspectives involve using computers to solve problems by breaking down complex tasks into simple steps. This approach helps us create efficient solutions and innovate. Learning these skills enhances problem-solving and critical thinking, preparing us for future careers.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

6.10.1 The significance of Computational Perspective

Computational perspectives are essential for students' future careers. As technology advances, computational thinking helps students solve complex problems efficiently. It promotes innovation, critical thinking, and adaptability, which are crucial in today's job market. By learning these skills, students can excel in various fields and stay competitive in a digital world.

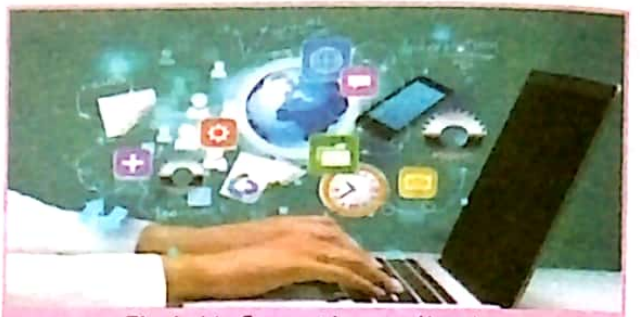


Fig.6.44: Computing application

6.11 Computing Applications

The need for collaborations to improve designs of computing applications is essential. When diverse teams work together, they combine their unique skills, experiences, and perspectives, leading to more innovative and user-friendly software. Collaboration fosters creativity and ensures that applications meet the needs of a wider audience. It also allows for resource sharing and more effective problem-solving, enhancing the overall quality of the final product.

6.11.1 The Importance of Collaboration in Enhancing Computing Application Design

Collaborations are essential for improving the designs of computing applications. By working together, diverse teams can combine their unique skills, experiences, and perspectives to create more innovative and user-friendly software. Here's why collaboration is crucial:

Diverse Perspectives:

Collaboration gathers people from diverse backgrounds, which brings a wider range of ideas and solutions. This diversity results in more creative and effective designs that appeal to a broader audience.

Enhanced Innovation:

When people from different fields collaborate, they can combine various technologies and methods, resulting in groundbreaking innovations in computing applications.

User-Centric Design:

Including end-users and stakeholders in the design process ensures the final product meets their needs and expectations. Feedback from different users helps create more user-friendly and accessible applications.

Resource Sharing:

Working together lets us share resources like what we know, the tools we have, and the money we can use. This sharing helps things move faster and makes the final product better.

Problem Solving:

Working together helps teams solve hard problems better. Different ideas and skills make

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

problem-solving stronger.

6.12 Resources for Equal Information Accessibility

Making things that deal with fairness in accessing information means creating tools or projects to give everyone equal access to information, no matter who they are or where they live. Here are some ideas:

Digital Libraries:

Create online libraries offering free access to educational materials like textbooks and articles. These libraries should be available to anyone with internet, helping to reduce the digital gap and encourage lifelong learning.

Fig.6.45: Digital Libraries:

Fig.6.46: Accessible Learning Platforms

Accessible Learning Platforms:

Create e-learning platforms that cater to people with disabilities. Include tools like screen readers, captions, and alternative content formats. Ensure the platforms meet various learning needs and preferences. Make sure everyone can access educational resources equally.

Community Information Centers:

Set up community information centers with computers, internet, and learning materials. Focus on underserved communities. These centers provide access to information, skill development, and social connections. Empower individuals to engage in the digital era fully.

Fig.6.47: Community Information Centers

Fig.6.48: Mobile Applications

Mobile Applications:

Create phone apps for important services like healthcare or job help. These apps should be easy to use, respectful of different cultures, and in many languages to help everyone.

Open Educational Resources (OER):

Make and share open educational resources for free. These resources can be used, changed, and shared by anyone. By doing this, we make sure more people can get knowledge and help teachers and learners adapt content to what they need.

Fig.6.49: Open Educational Resources (OER)

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Digital Literacy Programs:

Start programs that teach people how to use the internet wisely. They'll learn to find reliable info, use digital tools safely, and protect their privacy. These programs help people think carefully and get involved online.

Fig.6.50: Digital Literacy Programs

Fig.6.51: Partnerships and Collaborations

Partnerships and Collaborations:

Encourage teamwork between government, schools, non-profits, and businesses to fix problems in information access and make it fair for everyone. By working together, they can make big plans to help people who do not have enough access. These ideas can help everyone get the information they need for school, work, or anything else, no matter how much money they have or other problems they face.

6.13 Collaborative Tools

Collaborative tools are software applications designed to facilitate teamwork and communication among individuals who may be working remotely or across different locations. These tools provide features that allow team members to work together on projects, share information, and communicate effectively. Below is a summary of some well-known collaborative tools.

Google Drive:

Google Drive is a service for storing and sharing files online. It lets you save documents, spreadsheets, and presentations in the cloud. You can work together with others in real-time, editing files together and seeing changes as they happen.

Fig.6.52: Google Drive

Fig.6.53: Slack

Slack:

Slack is a messaging app where teams can communicate in channels based on topics or projects. You can send direct messages, share files, and connect with other tools and services.

Trello:

Trello is a tool for managing projects using boards, lists, and cards. You can make boards for different projects, add tasks as cards, and move them as they progress. It also lets teams collaborate by adding comments, attachments, and due dates.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

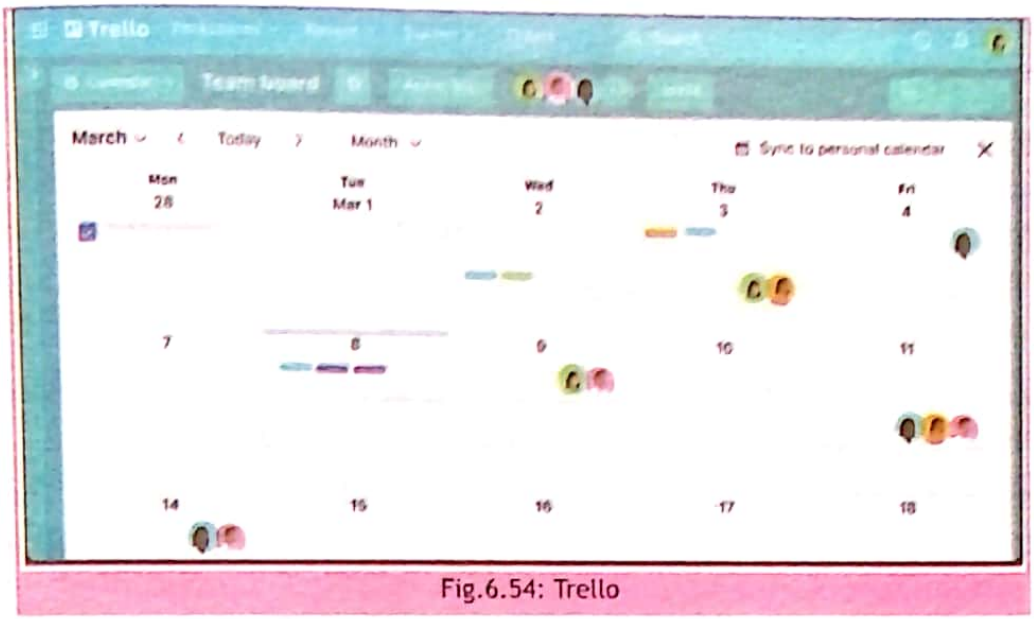


Fig.6.54: Trello

Microsoft Teams:

Microsoft Teams is a collaboration platform that integrates chat, video meetings, file storage, and application integration. It allows users to communicate and collaborate in real time, share files and documents, and work together on projects.

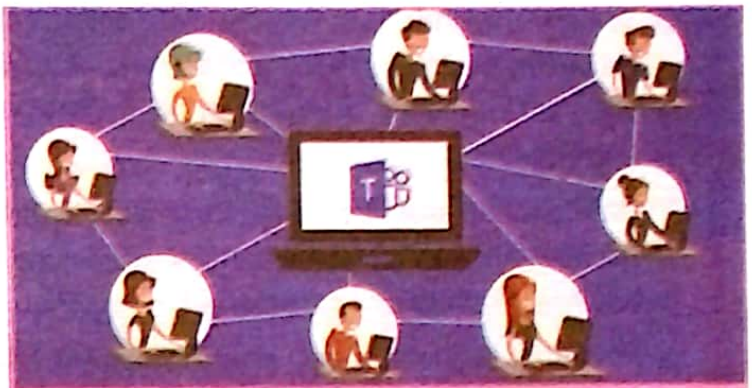


Fig.6.55: Microsoft Teams

Zoom:

Zoom is a video conferencing platform that enables virtual meetings, webinars, and conference calls. It offers features such as screen sharing, chat, and recording, making it suitable for remote collaboration and communication.

Fig.6.56: Zoom

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Confidentiality:** Security protocols, like encryption, keep data confidential. Encryption changes data into a format only accessible by authorized parties with decryption keys. This stops unauthorized individuals from accessing sensitive information.
- **Data Integrity:** Security protocols help maintain data integrity by ensuring that data remains unchanged during storage and transmission.
- **Authentication:** Security protocols use authentication to verify user or device identities. This ensures that only authorized people or systems access or send data.
- **Access Control:** Security protocols enable access control mechanisms. These controls decide who can access, change, or send data. It ensures only authorized users access data, systems, and places.
- **Secure Transmission:** Sending data online can be risky because hackers might try to spy. However, tools like "SSL/TLS" for websites and "VPNs" for remote connections encrypt your data, making it like a secret code that only you and the intended recipient can understand.
- **Data Backup and Recovery:** Data security protocols cover backup and recovery. Regular backups and secure storage rules make sure data can be brought back if it's lost due to hardware failure or a cyberattack.
- **Phishing:** Scammers may send fraudulent emails, texts, or messages that appear to be from trusted organizations. Scammers ask you to provide personal information like passwords or financial details.
- **Data Breaches:** When organizations or websites are attacked online, they might lose customer data, and if you're one of their customers, your data could be at risk too. Identity thieves might try to trick people into giving away personal information by pretending to be someone trustworthy.
- **Online Scams:** Scammers make fake websites or ads online to fool people into giving personal information or paying money to the wrong places.
- **Cyber-attacks:** Cyberattacks put both people and organizations at risk by targeting their security and privacy.
- **DDoS (Distributed Denial of Service) Attack:** A DDoS (Distributed Denial of Service) attack is when many infected computers work together to flood a website or system with too much traffic.
- **Ransomware:** Ransomware is a type of malicious software that locks or encrypts your files so you cannot access them.
- **Spyware:** Spyware is harmful software that secretly collects information about someone without them knowing.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- **Viruses:** Viruses are programs that copy themselves and infect a computer's files or software. They can spread to other systems when infected files are shared.
- **Phishing:** Phishing attacks deceive people into giving away sensitive details like passwords or credit card numbers by pretending to be a reliable source. These attacks usually use fake emails, websites, or messages.
- **Malware (Malicious Software):** Malware refers to harmful software like viruses, spyware, and Trojans, which can damage computer systems or steal information.
- **Two-Factor Authentication (2FA):** Two-factor authentication (2FA) enhances security by requiring two separate forms of identification to access a system, account, or app.
- **Biometric Verification:** Identification method using unique physical features, like fingerprints or facial recognition, for security purposes.
- **Symmetric Cryptography:** Encryption method using the same key for both encryption and decryption of data, known for its speed and efficiency.
- **Asymmetric Cryptography:** Encryption method using a pair of keys (public and private) for encryption and decryption, offering secure communication without sharing a secret key.
- **Caesar Cipher:** An ancient encryption method where each letter in the plaintext is shifted by a fixed number of positions in the alphabet.
- **Substitution Cipher:** Encryption method replacing each letter in the plaintext with a different letter according to a fixed system.
- **Transposition Cipher:** Encryption method rearranging the positions of characters in the plaintext without changing the characters themselves.
- **Vigenère Cipher:** Encryption technique using a keyword to shift letters in the plaintext, providing a more complex substitution pattern.
- **Transport Layer Security (TLS):** Cryptographic protocol used to secure internet communication, ensuring data confidentiality and integrity.
- **Secure Sockets Layer (SSL):** Predecessor to TLS, used for securing data transmission over the internet.
- **Internet Protocol Security (IPsec):** Protocol suite for securing IP communications by authenticating and encrypting each IP packet.
- **Secure Shell (SSH):** Protocol providing secure remote access to servers and network devices, offering strong authentication and encrypted communication.
- **Hypertext Transfer Protocol Secure (HTTPS):** Extension of HTTP used for secure communication over a computer network, encrypting data exchanged between a web browser and a website.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- **Phishing Attack:** Cybersecurity threat involving tricking individuals into giving away sensitive information through fake emails, websites, or messages.
- **Digital Libraries:** Online repositories offering free access to educational materials like textbooks and articles to reduce the digital gap.
- **Accessible Learning Platforms:** E-learning platforms designed to cater to people with disabilities, including tools like screen readers and captions.
- **Community Information Centers:** Centers providing access to computers, the internet, and learning materials, especially in underserved communities.
- **Open Educational Resources (OER):** Free, accessible educational materials that can be used, modified, and shared by anyone.
- **Google Drive:** Cloud storage service allowing real-time collaboration on documents, spreadsheets, and presentations.
- **Slack:** Messaging app facilitating team communication through channels based on topics or projects.
- **Trello:** Project management tool using boards, lists, and cards to organize tasks and facilitate team collaboration.
- **Microsoft Teams:** Collaboration platform integrating chat, video meetings, file storage, and application integration for real-time teamwork.
- **Zoom:** Video conferencing platform enabling virtual meetings, webinars, and conference calls, with features like screen sharing and recording.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. What is the primary purpose of security protocols?
   a. To speed up data transmission
   b. To protect data and prevent unauthorized access
   c. To lower the cost of data transmission
   d. To improve the appearance of web pages
2. What technique is used by security protocols to maintain confidentiality?
   a. Authentication b. Encryption c. Data backup d. Data recovery
3. Which method ensures data integrity during transmission?
   a. Encryption b. Hashing and digital signatures
   c. Phishing d. File sharing
4. Which of the following verifies user or device identity?
   a. Access control b. Encryption c. Authentication d. Data recovery
5. What do security protocols use to control who can access or modify data?
   a. Authentication b. Access control c. Malware protection d. Cryptography
6. What type of software protects against malware and viruses?
   a. Digital signatures b. Firewalls and anti-malware tools
   c. Access control mechanisms d. SSL/TLS
7. Which of the following is an example of a type of cyberattack mentioned in the text?
   a. Authentication b. Ransomware c. Access Control d. Hardware failure
8. What does Two-Factor Authentication (2FA) typically combine?
   a. A password and encryption b. A password and a secondary form of ID
   c. An IP address and encryption d. A virus scan and a firewall
9. What is an example of asymmetric cryptography?
   a. Sharing a common key for encryption
   b. Using public and private keys
   c. Using the same key for encryption and decryption
   d. Changing data transmission speed
10. Which security protocol is commonly used to secure internet communication?
    a. TLS/SSL b. FTP c. UDP d. HTTP
11. Which cipher shifts each letter by a fixed number in the alphabet?
    a. Vigenère cipher b. Transposition cipher c. Caesar cipher d. Substitution cipher

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

12. Which security protocol is used for secure remote access?

a. Ipsec b. SSH c. HTTP d. FTP

13. What is phishing?

a. A type of encryption method

b. An attack where fake emails trick users into giving sensitive information

c. A method to secure data transmission

d. A secure file transfer protocol

14. Which best practice can help prevent identity theft?

a. Sharing passwords with trusted individuals b. Using strong, unique passwords

c. Storing all sensitive information in the cloud d. Ignoring financial statements

15. What does cryptography protect?

a. Physical hardware b. Data and communication

c. Marketing strategies d. System updates

16. Which of the following is an example of a security method used to enhance user authentication?

a. DDoS attack b. Spyware

c. Biometric verification d. Phishing

17. Which of the following tools is commonly used to support online collaboration in computing projects?

a. Photoshop b. Google Drive

c. Excel d. VLC Media Player

Give short answers to the following Short Response Questions (SRQs).

1. Why is it important to apply security measures when using online platforms?
2. What are the main functions of security protocols in today's digital world?
3. What are the key reasons for using security protocols to protect data?
4. How can sharing private information lead to identity theft?
5. What are the benefits and potential privacy issues of biometric verification?
6. What does Two-Factor Authentication (2FA) require for user identification?
7. Name one advantage and one disadvantage of Biometric Verification.
8. What is the primary purpose of cryptography in data transmission?
9. Describe the Caesar Cipher in brief.
10. What protocol is used for secure communication over the internet?
11. Why is it important to understand how and when your information is collected and used online?

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

12. Why is collaboration important in the design of computing technologies, especially when addressing equity in access to information?

Give long answers to the following Extended Response Questions (ERQs).

1. How do security protocols ensure data protection in today's digital world, and what specific techniques and measures do they employ to maintain confidentiality, and authentication?
2. What are the primary risks associated with sharing private information online, and how can individuals protect themselves against identity theft?
3. How do different types of cyber-attacks, such as DDoS, ransomware, spyware, viruses, and phishing, threaten the security and privacy of individuals and organizations?
4. Discuss the process and benefits of Biometric Verification, as well as the privacy concerns associated with it.
5. Describe how cryptography ensures the safe transmission of data, detailing the difference between symmetric and asymmetric encryption.
6. Compare and contrast the different types of cryptographic methods mentioned, including their use cases and efficiency.
7. Imagine you are developing a new mobile application that stores user data. Explain how you would ensure the security of the data during both storage and transmission.
8. You are working with a diverse team to develop a new educational website. Describe how you would use online collaborative tools and communication strategies to ensure all team members can contribute equally.

Activity 1: Identify and Apply Safe Practices

Objective: Help participants understand and implement safe practices for collaborating on digital platforms.

Materials Needed:

- Internet-connected devices
- Access to an online collaboration tool (e.g., Google Drive)

Instructions:

- Briefly explain safe practices for online collaboration, such as using strong passwords and recognizing phishing attempts.
- In small groups, have participants identify three safe practices and apply them to an online collaboration tool.
- Each group will present their implemented practices and explain their importance.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Activity 2: Discuss Security Threats and Mitigation

Objective: Educate participants on security threats and mitigation strategies.

Materials Needed:

- Handouts or digital guides on 2FA, biometric verification, and secure data transmission

Instructions:

- Briefly discuss common security threats, including phishing, malware, and data breaches.
- Divide participants into groups and assign each group one threat to focus on.
- Groups will research mitigation strategies such as 2FA, biometric verification, or secure data transmission, and present their findings to the class.

Activity 3:Collaborate on Strategies for Equity and Access

Objective: Develop strategies to ensure equitable access to information on digital platforms.

Materials Needed:

- Whiteboard or online collaborative tool (e.g., Jamboard)

Instructions:

- Discuss the importance of equity and equal access to information.
- In small groups, brainstorm strategies to provide equitable access, such as ensuring all participants have the necessary technology and skills.
- Each group presents their strategies, and the class discusses how to implement them in various digital collaboration scenarios.

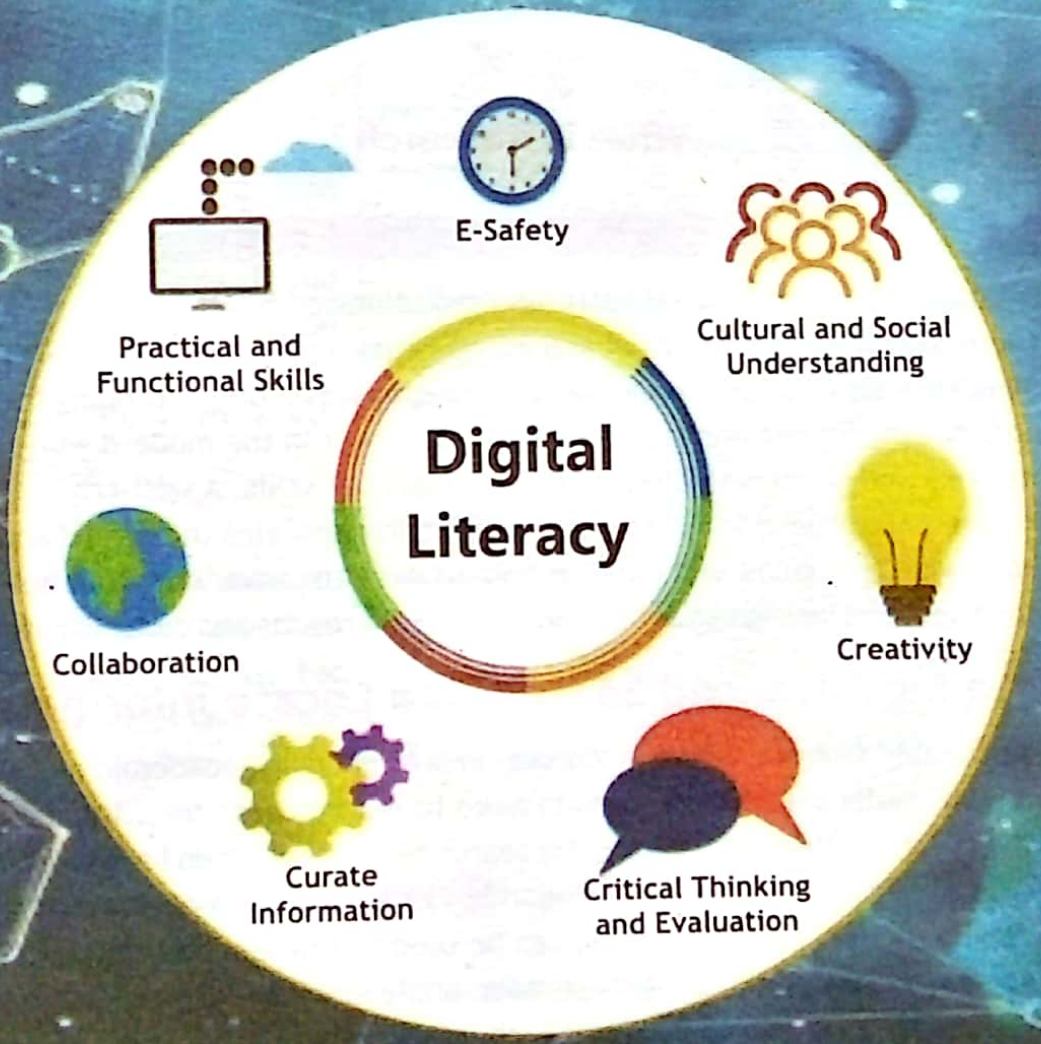
Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

7 Digital Literacy

Learning Outcomes

At the end of this unit students will be able to:

- create an artefact that answers a research question, communicates results and conclusions through digital resources or tools



Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Introduction

Research is the process of investigating a specific question or topic by systematically gathering, analyzing, and interpreting information. It is an essential skill for students, as it helps you to solve problems and answer questions effectively. At the core of any research project is the research question, which serves as a clear, focused, and specific guide for the entire investigation. A well-defined research question helps narrow down the scope of the research and provides direction.

Fig.7.1: What is Research?

In today's digital age, the use of digital tools has revolutionized how research is conducted and presented. These tools allow you to gather and process data more efficiently, while also enabling you to communicate your findings in creative and engaging ways. Understanding the importance of these tools is critical for becoming an effective researcher in the modern world. Conducting research effectively and communicating results are essential skills. A well-crafted artefact not only addresses a research question but also effectively communicates the results and conclusions through digital tools. This guide will explore how to perform advanced searches for locating research information, and how to use digital tools to present results and conclusions.

7.1 Performing Advanced Searches to Locate Information

Information is available from a variety of sources, including books, academic journals, articles, websites, and multimedia platforms. Students need to learn how to identify and use credible sources of information for their research. Basic search techniques often involve entering simple keywords into search engines, but advanced searches require more sophisticated methods. For instance, Boolean operators (AND, OR, NOT) can be used to refine search results. Students can also use quotation marks to search for exact phrases, apply filters to narrow down by date or type of source, and utilize academic databases such as Google Scholar and PubMed to access specialized and peer-reviewed materials.

Fig 7.2 shows how to choose the Advanced Search option on Google home page, after typing your search string click on Tools Option and Advanced Search Menu will open.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

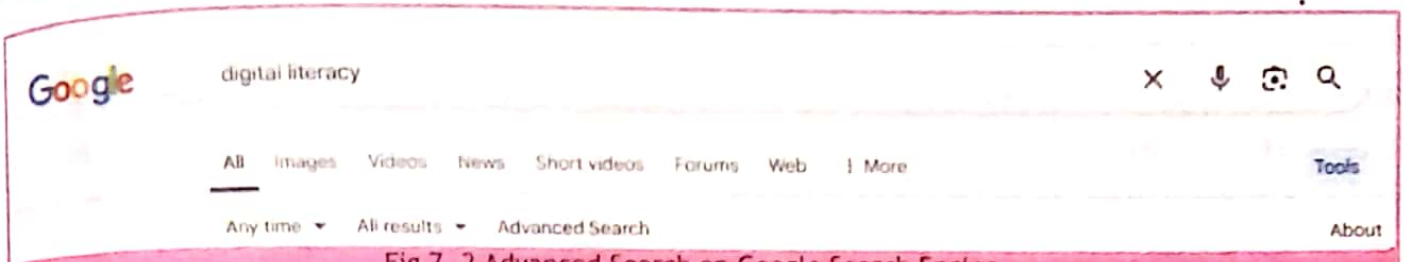


Fig 7. 2 Advanced Search on Google Search Engine

Fig 7.3 shows the options when you select the option of advanced search on Google. (The interface may vary depending on the type and version of your web browser) In the "Search for" option you can select various options like everything if you are searching for common information. You can choose Articles if you are searching for articles related to your search string.

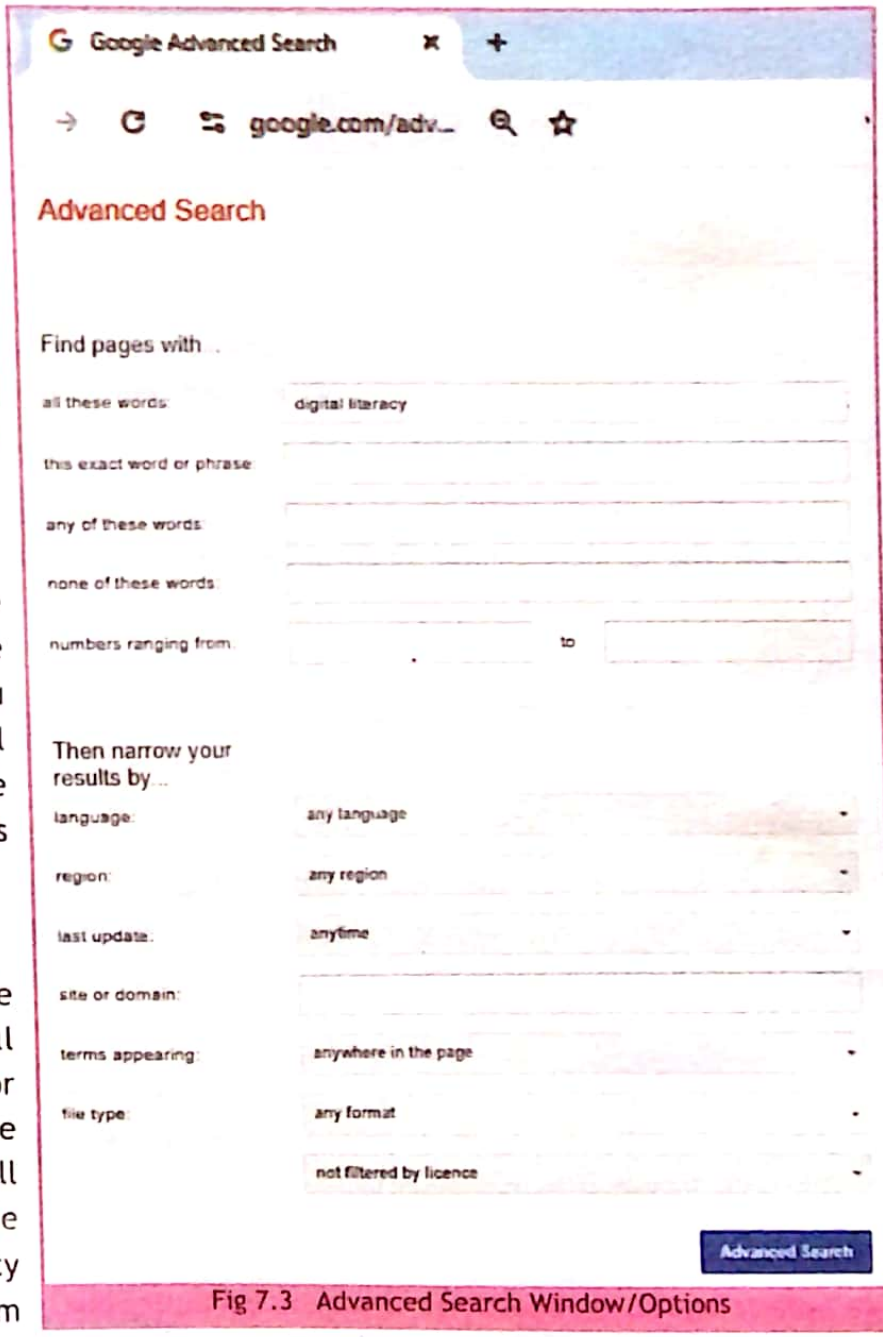


Fig 7.3 Advanced Search Window/Options

All these words:

This field is used to include multiple words that must all appear in the search results. For example, if you want to search for the string Digital Literacy Google will return the results that include both words Digital and Literacy.

This exact word or phrase:

This field is used to include multiple words or phrases that must all appear in the search results. For example, if you want to search the string "Digital Literacy" Google will return the results that include complete phrase Digital Literacy rather than separate words from the phrase or search string. (Tip: double quotes can be used to exactly find the search string)

Any of these words:

This field is used to include any of the words from the search string. For example, if you want to search for the string Digital Literacy Google will return the results that include some results which will include the word Digital while some results which contain the word Literacy.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

None of these words:

This allows you to exclude specific words from your search results. For example, "Digital - Literacy" will show results that have "Digital" but not "Literacy". (Tip: Hyphen - sign can be used before the word to exclude the words to be excuded)

Numbers range:

This is useful for finding results within a specific range (e.g., dates, prices). For example, "books Rs500..Rs1500" will show books priced between Rs.500 to Rs. 1500.

Language:

You can filter search results based on language preferences. For example, choose "Urdu" to limit results to Urdu language content.

Region:

You can limit results to specific countries or regions. Fo r example, you can specify country name "Pakistan" to only get results from Pakistan based websites.

Site or domain:

Site operator limits the search to a specific website or domain. For example, "site: fbise.edu.pk" will return the results from the Federal Board of Intermediate and Secondary Education website only.

File type:

You can search for specific file types like PDFs, Word documents, PowerPoint files, etc. for example, "filetype: pdf digital literacy" will return only PDF files about digital litracy.

Usage rights:

This option will used to filter results based on usage rights, such as free content which does not restrict copyright policy. For example, to search images if you write "labeled for reuse" it will return the results/images which you can use for free.

Intitle:

This option will return the web pages that contains the keywords/search string in the title of document/page. For example, Intitle: HSSC Result 2025 will return all the webpages that will contain HSSC Result 2025 in their title.

Practice Examples:

Example 1: Searching for articles on machine learning but excluding "deep learning".

Search: machine learning -deep learning

Example 2: Searching for PDFs about Python programming.

Search: filetype: pdf python programming

Evaluating the credibility of sources is crucial for ensuring that research findings are reliable. Credible sources include peer-reviewed articles, websites with trusted domains (such as .edu,

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

.gov, or .org), and up-to-date information from well-known authors or institutions. Once the relevant information is found, students can use citation management tools like Zotero or Mendeley to organize their sources and keep track of important references. This will help ensure that all sources are properly cited in the final research artefact.

7.1.1 Creating the Artefact

To begin creating the artefact, the first step is data collection by performing advanced searches to locate credible sources. This involves using search engines, academic databases, and digital libraries to find relevant information related to the research question. Various primary and secondary methods of data collection have been discussed in detail in previous grades. You can use any reliable method of data collection. The reliability and accuracy of the data is a challenge. Therefore, while collecting data, you should be able to verify a valid source of data.

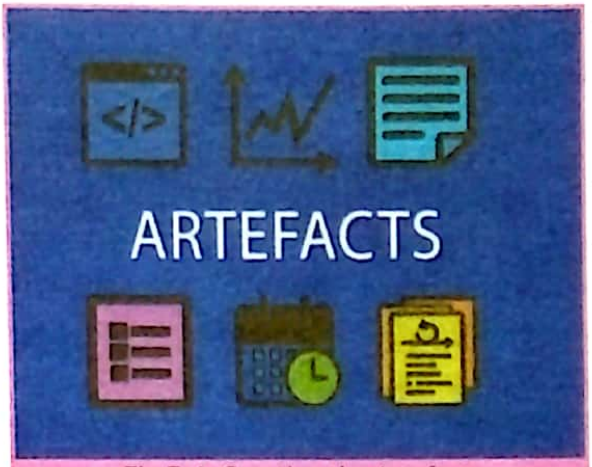


Fig.7.4: Creating the Artefact

7.1.2 Communication of Collected Data

Once the data has been collected and analyzed, the next step is to communicate the findings through digital tools. There are a wide variety of digital tools available, each suited to different types of presentations. For text-heavy reports, tools like Microsoft Word, Google Docs, or LaTeX can be used to write and format research papers. These tools allow for detailed explanations of findings, along with proper referencing and citations. For visual presentations, PowerPoint, Google Slides, or Prezi are commonly used. These platforms enable students to create visually engaging slideshows that communicate key points in an organized manner. Data visualization tools like Excel, Google Sheets, and Tableau can be used to create charts, graphs, and tables that help illustrate trends or patterns in the research data.

Fig.7.5: Communication of Collected Data

7.1.3 Layout of the Digital Artefact

The artefact must be structured in a logical way to effectively communicate results and conclusions. Following is a list of some important parts of a digital artefact like a sale report, a software user manual or a presentation etc.

Introduction

Typically, a digital artefact should start with the section named Introduction, which presents the

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

research question and objectives, a methodology section explaining how the research was conducted, a results section that presents the data using visuals where appropriate, and a conclusion that summarizes the key findings. Digital tools allow for the inclusion of interactive elements, multimedia, and hyperlinks, all of which can enhance the depth and accessibility of the artefact.

Assume you are designing a digital artefact about the impact of social media on mental health. The introduction part can contain a brief overview of the task/project. It includes some background information, and context of the purpose for which you are designing this artefact.

Findings

An important aspect of creating the artefact is ensuring that the findings are communicated clearly and concisely. Visuals play a key role in this process, as they help to break down complex data into more digestible formats. Well-designed graphs, charts, infographics, or videos can make it easier for an audience to understand the significance of the results. Students must be careful to ensure that all visuals are clearly labeled, accurate, and relevant to the data being presented. Visual aids should complement the research rather than distract from it.

For example, you collected some data from the audience by using questionnaires. The data show different ratios and nature of the impact of social media according to different age groups. You can organize your findings in tabular form.

Result/Conclusion

Finally, the artefact must be concluded by providing a clear interpretation of the results and their implications. Conclusions should directly address the research question and provide answers or insights based on the data collected. It is also important to suggest future research possibilities or areas where further investigation could be beneficial. For example, if the research focused on the effects of a school breakfast program on student attendance, the conclusion might highlight specific grade levels or demographic groups that showed the most improvement and suggest expanding the study to other variables such as academic performance.

Creating an artefact to answer a research question has several steps. First, you need to formulate a clear question. Then, you conduct advanced searches to gather reliable data. Next, you use digital tools to organize and present your findings. Finally, you ensure that your artefact effectively communicates your results. This process helps you develop research, analytical, and digital skills. It also enables you to present complex ideas in a clear and engaging way. By leveraging digital resources and tools, you can create impactful artefacts that showcase your understanding and share your conclusions with others.

7.1.4 Using Digital Tools to Communicate Results and Conclusions

There are many digital tools available to students for presenting their research. Text-based tools such as Microsoft Word or Google Docs are useful for writing detailed research reports, while presentation tools like PowerPoint, Google Slides, or Prezi are ideal for visual presentations. When dealing with data, tools such as Excel, Google Sheets, or Tableau can be used to create graphs, charts, and tables that help illustrate key findings. For more visually engaging

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

presentations, multimedia tools like Canva and Piktochart can be used to create infographics, while collaboration tools like Google Drive, Trello, or Microsoft Teams enable groups to work together efficiently.

Fig.7.6: Digital tools available to students

Choosing the right digital tool depends on the nature of the research and the intended audience. For example, if a research project involves extensive numerical data, it may be better to use data visualization tools like Excel or Tableau. On the other hand, for more general presentations, PowerPoint or Google Slides are often more appropriate. Regardless of the tool used, students should structure their artefact clearly, with sections such as an introduction, which presents the research question and objectives, a methodology section describing the research process, results with clear visuals like graphs and charts, and a conclusion which summarizes key findings and suggests future research directions. Proper citation of all sources is necessary, and digital tools like EasyBib or Zotero can be helpful in generating citations in styles such as APA, MLA, or Chicago.

Creating effective visuals is an important skill, as graphs, charts, and infographics help to clarify and emphasize key points in the research. Students should ensure that all visuals are properly labeled and directly related to their findings. Visuals should enhance the understanding of the audience, not overwhelm them with unnecessary Information.

7.1.5 Creating a Research Artefact

To reinforce the concepts learned, we can create a research artefact. For this project, we will select a research question of interest, for example, "How does social media affect student productivity?". You can then apply advanced search techniques to locate credible information on their topic, organize their findings, and use a digital tool to create an artefact that communicates their research.

Figure 7.2 shows the process of creating a research artefact. At the first step i.e. initial problem, identify a problem or research question. e.g. How can we reduce energy consumption in commercial buildings? To explain the problem, clarify and refine the problem statement. e.g. What are the most effective strategies for reducing energy consumption in commercial office buildings in urban areas? Now the explicated Problem will be the final problem statement i.e. This artefact will investigate the effectiveness of retrofitting commercial office buildings with energy-efficient lighting and HVAC systems.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

In the next step create an outline and define the requirements for the artefact. i.e. introduction, literature review, case studies, recommendations, conclusion. The requirements can be Interactive dashboard, 3D model, references, and presentation. In the next step list the requirements. i.e. Visually appealing, easy to navigate, accurate, reliable and relevant content. The technical requirements will be Interactive elements, compatible with multiple devices.

To design and develop the artefact, create the artefact based on the outline and requirements. i.e. design an interactive dashboard using power bi, create a 3d model using sketch up, write the content and references. The final product will be the desired artefact. An interactive dashboard presenting energy consumption data and 3D models of commercial office buildings. To demonstrate the artefact, present it to the audience. To evaluate it, assess the effectiveness and quality of the artefact. Evaluate the artefact based on clarity, organization, and effectiveness of presentation. The evaluation has identified areas for improvement, such as adding more interactive elements or clarifying the recommendations.

As part of the activity, students can work in groups to collaboratively research different questions, gather data, and present their findings using tools such as Google Slides or Prezi. This allows them to practice using digital tools, while also fostering teamwork and collaboration. After completing the artefact, each group will present their work to the class, receiving peer feedback on the clarity, organization, and effectiveness of their presentation.

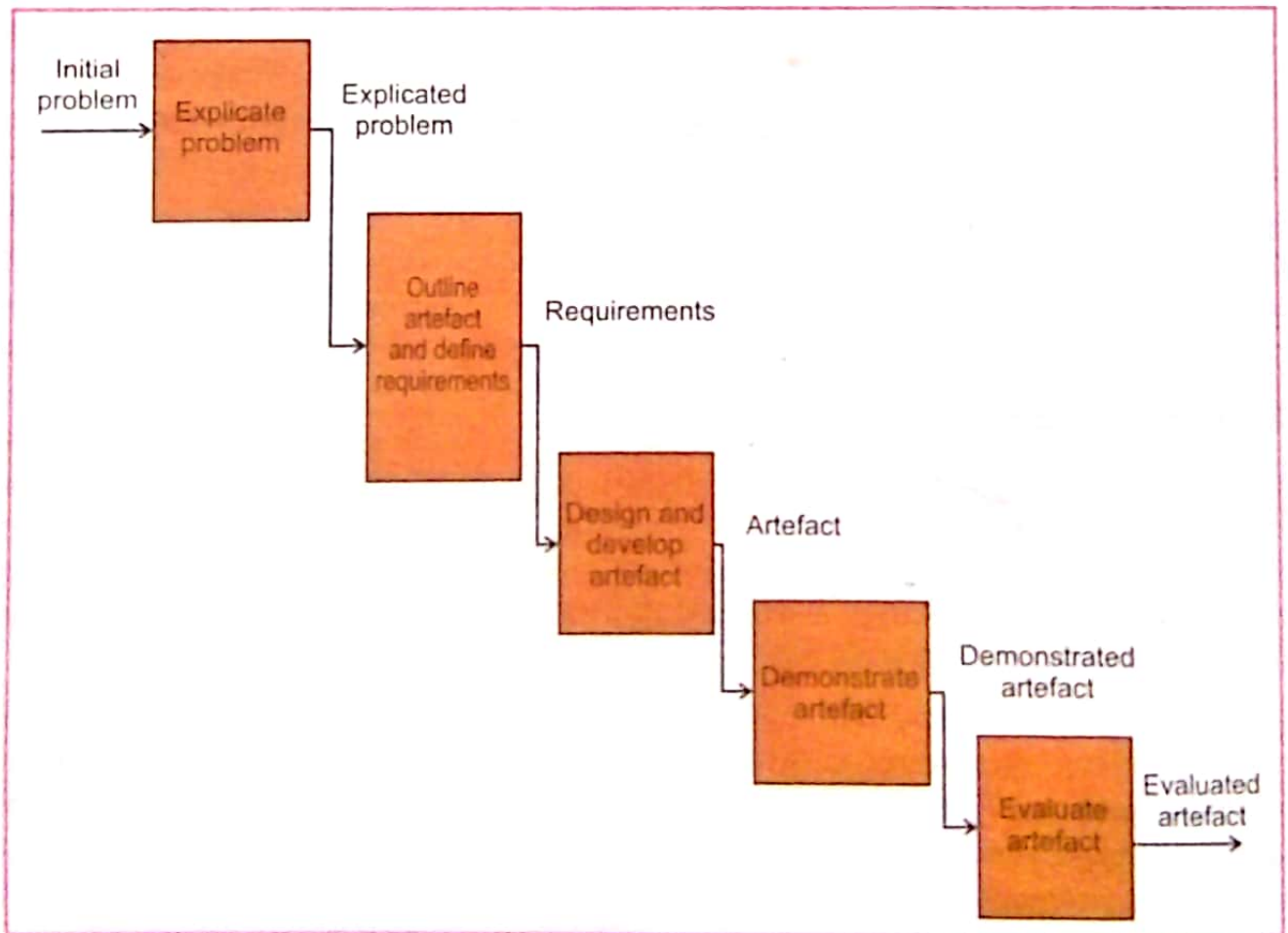


Fig.7.7: Process of creating artefacts

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

7.1.6 Reflection and Self-Assessment

After completing the research project, it is important to reflect on the research process. This reflection allows us to identify challenges, such as difficulty finding credible sources or presenting complex data, and to consider how to overcome these challenges. You can also think about what worked well in your research and where they might improve in future projects.

In addition to reflecting on the research process, you should also assess your ability to use digital tools effectively. A self-assessment checklist can help you evaluate your skills in selecting appropriate tools, organizing information, and presenting data in a clear and engaging way. This reflection and self-assessment will help to build confidence and improve research and presentation skills over time.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Research** is the process of investigating a specific question or topic by systematically gathering, analyzing, and interpreting information. It is an essential skill for students, as it helps you to solve problems and answer questions effectively.
- **Research questions** are at the of any research project, which serves as a clear, focused, and specific guide for the entire investigation.
- properly cited in the final research artefact.
- **Creating the artefact**, involves the first step of data collection by performing advanced searches to locate credible sources.
- **Sources of reliable data** include academic databases, and digital libraries to find relevant information related to the research question. Various primary and secondary methods of data collection have been discussed in detail in previous grades
- **A digital artefact** must be structured in a logical way to effectively communicate results and conclusions. Following is a list of some important parts of a digital artefact like a sale report, a software user manual or a presentation etc.
- **Findings** An important aspect of creating the artefact is ensuring that the findings are communicated clearly and concisely. Visuals play a key role in this process, as they help to break down complex data into more digestible formats.
- **Well-designed graphs, charts, infographics, or videos** can make it easier for an audience to understand the significance of the results.
- **Result/Conclusion** Finally, the artefact must be concluded by providing a clear interpretation of the results and their implications. Conclusions should directly address the research question and provide answers or insights based on the data collected
- **Digital Tools** available to students for presenting their research. Text-based tools such as Microsoft Word or Google Docs are useful for writing detailed research reports, while presentation tools like PowerPoint, Google Slides, or Prezi are ideal for visual presentations.
- **Creating effective visuals** is an important skill, as graphs, charts, and infographics help to clarify and emphasize key points in the research. You should ensure that all visuals are properly labeled and directly related to their findings. Visuals should enhance the understanding of the audience, not overwhelm them with unnecessary information.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. Which of the following advanced search operators would be used to limit the results to a specific website domain such as .gov, or .edu
   a) Quotation marks (" "_)
   b) Filetype,
   c) hyphen Operator (-)
   d) Site Operator
2. What is the primary benefit of using hyphen operators in advanced searches?
   a) It narrows down result to academic articles only
   b) It ensures you only get peer reviewed sources.
   c) It helps include or excluding specific terms from search results
   d) It automatically filters irrelevant websites.
3. Which digital tool is the best suited for performing large data of complex numerical data in an interactive format.
   a) Google slides
   b) Tableau
   c) Canva
   d) Adobe Spark
4. What is the first step in creating an artefact?
   a) Data Analysis
   b) Data Collection
   c) Communication of Findings
   d) Verification of sources
5. What is the purpose of data visualization tools?
   a) To create a visually engaging slide show
   b) To write and format research papers
   c) To illustrate trends or patterns in research data
   d) To verify valid sources of data
6. What is the most effective way to structure a digital research presentation of clarity?
   a) Start with detailed statistics and end with general ideas
   b) Use visuals exclusively without any text
   c) Begin with the research question, followed by methodology, results, discussion and conclusion.
   d) Use a random order of topics to keep the audience engaged.
7. Which of the following would be the best strategy to find peer-reviewed journal articles on a specific research topic?
   a) Use Boolean operator with a file type search
   b) Limit your search to .com sites
   c) Search for your topic using the OR operator to get broader results
   d) Use "quotation marks" and the site .edu operator.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

8. How does the precision of a research question influence data collection methods?
   a. It limits the number of sources used
   b. It directs the selection of relevant, credible data
   c. It simplifies data visualization techniques
   d. It eliminates the need for secondary sources
9. Why might improper citation damage the effectiveness of a digital artefact?
   a. It keeps the artefact short and brief
   b. It questions the authenticity and reliability of the research
   c. It complicates the artifact's structure
   d. It distracts from the visuals used

Give short answers to the following Short Response Questions (SRQs).

1. How can Boolean operators enhance the effectiveness of online search?
2. Describe one advantage of using the site operator in advanced search queries.
3. How can a digital artefact be designed or created?
4. List all the steps to create an artefact.
5. What is the importance of layout of digital artefacts?
6. Why is the structure of a research presentation important when communicating results?
7. What is the main benefit of using infographics to present research findings?
8. Explain how advanced search techniques can save time in the research process.
9. What digital tool would you recommend for creating a video summary of research findings?
10. How do infographics simplify complex ideas for a general audience?
11. What is one key difference between static and interactive data visualizations?
12. In what ways can poorly designed visuals distort the intended meaning of research findings in an artefact?
13. Explain how the alignment between research questions, findings, and conclusions ensures consistency in a research artefact.

Give long answers to the following Extended Response Questions (ERQs).

1. Explain the role of advanced search techniques in improving the quality of research. Provide examples of specific operators and how they can be applied to a research topic.
2. Analyze the importance of structuring a research presentation effectively. How do introduction, methodology, results, discussion, and conclusion sections contribute to clear communication?
3. How can digital tools be used to engage diverse audiences in understanding complex research topics? Give examples of how visuals, videos, and interactive elements contribute to audience engagement.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

4. Explain how advanced search operators like site:, filetype:, and quotation marks can be used together to locate high-quality academic sources for a research paper. Provide a step-by-step example.
5. Evaluate the use of data visualization tools like Tableau and Microsoft Excel in presenting research findings. How do these tools help in interpreting and presenting complex datasets?
6. How do advanced search techniques improve both the accuracy and efficiency of locating information online? Discuss the importance of using the right keywords and operators for specific research tasks.
7. Analyze the benefits of creating an interactive dashboard for communicating research results. How does this tool differ from traditional static presentations?
8. Propose a strategy for peer-reviewing digital research artefacts in a classroom setting. How can peer feedback improve both content quality and presentation design?
9. Critically evaluate how the credibility and bias of sources affect the quality of a digital artefact. How can a researcher minimize bias while selecting sources?

Activity 1: Create a digital Research Artefact using advanced search technique

Objective: Students will use advanced search techniques to gather relevant information on a research topic and create a digital artefact to present their findings using appropriate digital tools.

Outcome: This activity will help students understand how digital research is helpful in creating artefact.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

8 Entrepreneurship in Digital Age

Learning Outcomes

At the end of this unit students will be able to:

- create a minimum viable product for their business

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Introduction

In today's digital world, the Internet makes it easy and affordable for entrepreneurs to test business ideas. Websites like eBay or personal business sites allow you to quickly see how customers respond to your products.

One helpful method is the Lean Start-Up approach. It focuses on building a **Minimum Viable Product (MVP)** – the simplest version of your product that still solves a problem for your customers.

Example:

When Dropbox started, the founders didn't build the full product. Instead, they made a short video demo showing how the app would work. Thousands of people joined their waitlist. This early feedback proved their idea had strong demand.

By testing with real users and improving based on feedback, you can build a successful product step by step. This keeps your idea customer-focused and cost-effective.

Key Takeaways

- Use online tools to test your business idea.
- Start with an MVP – a simple version of your product.
- Use customer feedback to improve your idea.
- Focus on solving a real problem for your users.

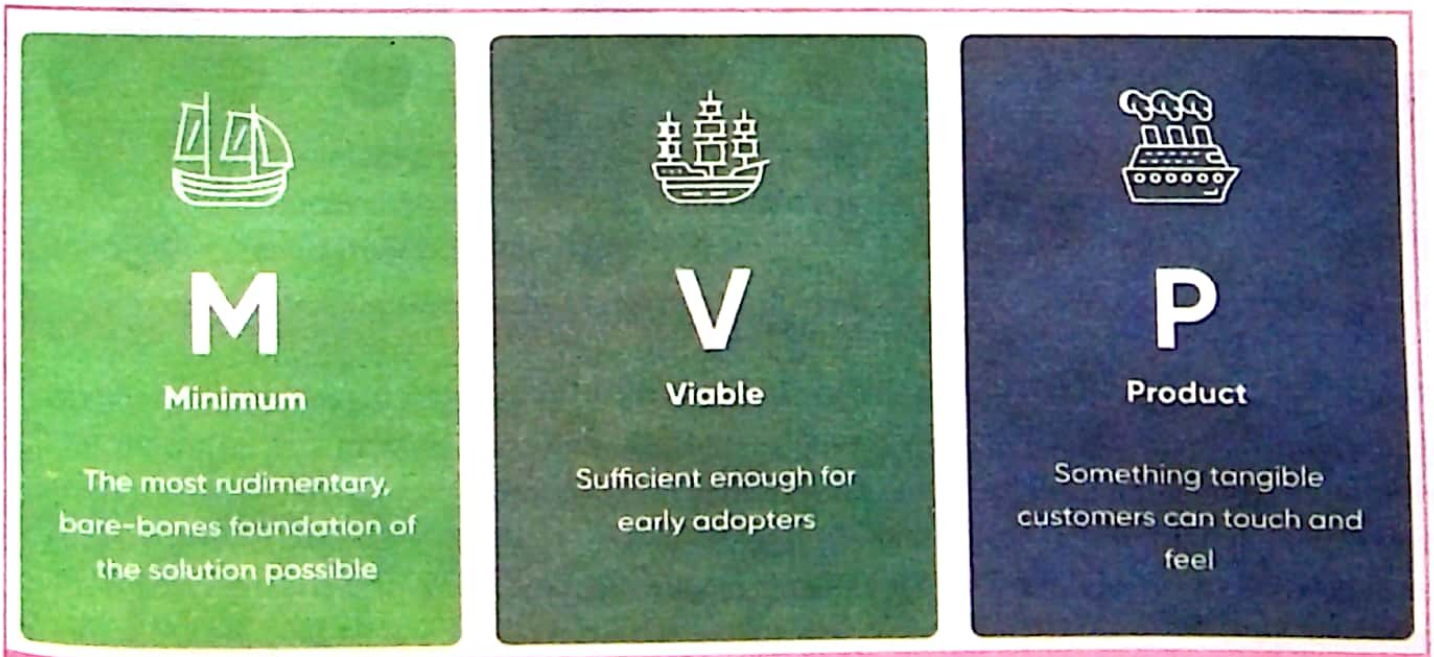


Fig.8.1: MVP

8.1 Building and Launching the MVP (Minimum Viable Product)

8.1.1 Developing the MVP

Minimum Viable Product (MVP) is a critical concept in product development, representing the

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

most basic version of your product that includes essential features for addressing the core problem or need of your target audience. Here are key points to ponder in developing the MVP:

Feature Prioritization: Identify the core features and functionalities that are absolutely necessary to address the primary problem or need. These features should provide value to early users.

Simplicity: Keep the MVP as simple and lean as possible. Avoid feature creep and focus on delivering a streamlined user experience.

Rapid Development: Utilize agile development methodologies to build the MVP quickly. This allows for flexibility in responding to changing requirements and user feedback.

Testing: Continuously test the MVP with real users to gather feedback and validate assumptions. Iterate and make improvements based on user insights.

Scalability: While the MVP is minimal, design it in a way that allows for scalability in the future. Ensure that the architecture and technology choices support future enhancements.

User Onboarding: Pay attention to user onboarding and guidance within the MVP. Make it easy for users to understand how to use the product effectively.

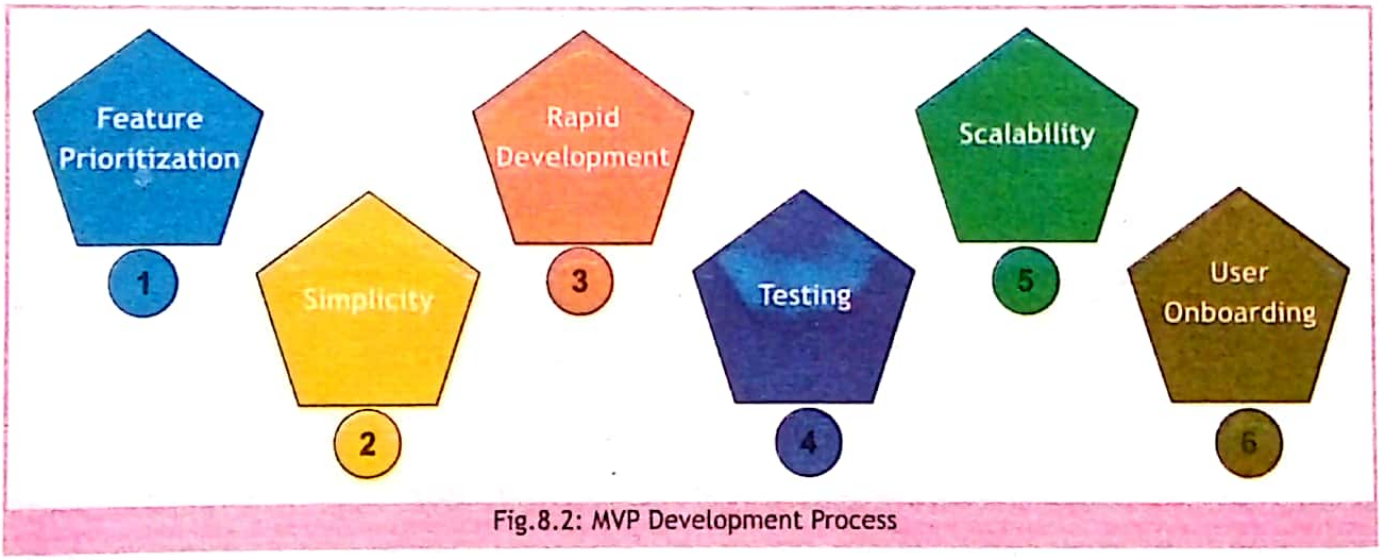


Fig.8.2: MVP Development Process

8.1.2 Launching Strategies and Considerations

Launching an MVP (Minimum Viable Product) is about carefully planning how to introduce your product to the market and acquire users. Here's how each strategy and consideration can be applied with examples to ensure the success of your MVP:

1. Targeted Audience

- **Strategy:** Identify a specific segment of your target audience that aligns best with your MVP's value proposition. Focus your initial marketing efforts on this group.

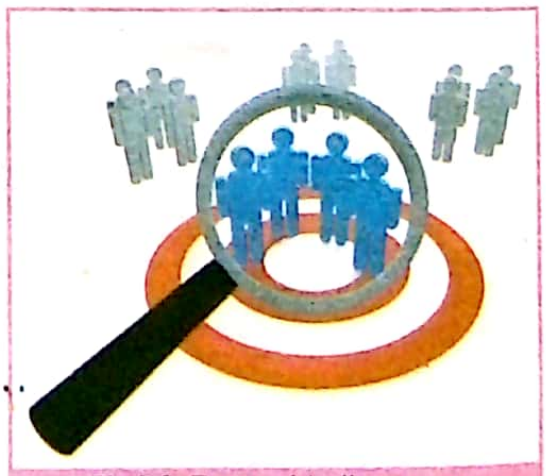


Fig.8.3: Targeted Audience

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

- **Example:** If you're launching a fitness app designed for new mothers, your targeted audience would be women who have recently given birth and are looking for postpartum fitness solutions. Instead of targeting all women or fitness enthusiasts, focus your marketing campaigns on this specific demographic through platforms like Instagram and parenting blogs that cater to this group. This helps ensure that the product resonates deeply with the users who are most likely to benefit from it.

2. Soft Launch

- **Strategy:** Consider a soft or limited launch initially to gather feedback from a smaller user base and make improvements before a broader release.

Fig.8.4: Soft Launch

- **Example:** Airbnb used a soft launch when it first introduced its platform. They initially offered their service only in San Francisco, allowing the team to refine the user experience, handle logistical issues, and build trust with both hosts and guests before scaling to other regions. This helped them test their MVP with real users without overwhelming their resources or reputation.
- **Approach:** A soft launch could involve limiting access to a specific city or group, and in the case of a mobile app, only releasing it to users who sign up for early access or join an invitation list. This helps iron out bugs, optimize the user experience, and understand user behavior before scaling up.

3. Beta Testing

- **Strategy:** Invite a select group of users to participate in beta testing to uncover potential issues, generate early user reviews, and gather testimonials.

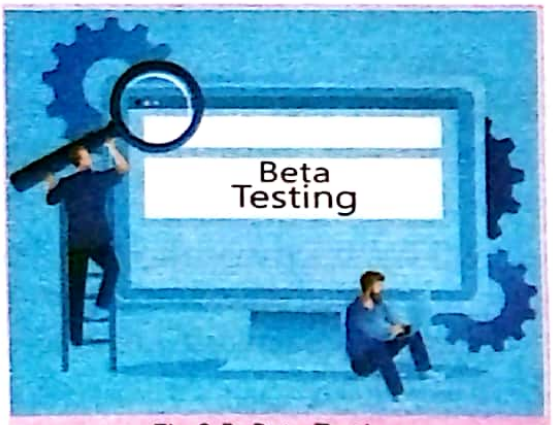


Fig.8.5: Beta Testing

- **Example:** Dropbox initially ran a private beta test before launching to the public. They invited a small number of users (mostly tech-savvy individuals) to test the platform and provide feedback. This group was instrumental in identifying bugs, suggesting improvements, and refining the product's features. In exchange, users were provided with extended free storage as a reward.
- **Approach:** You can run a closed beta for a small set of users, offering them early access in exchange for detailed feedback. For example, a new e-commerce platform could send invites to a few hundred users, track their behavior on the site, and get detailed feedback about navigation, checkout experience, and product offerings.

4. Marketing and Promotion

- **Strategy:** Develop a marketing plan that includes strategies for creating awareness,

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

generating interest, and driving user acquisition through digital marketing, social media, and content marketing.

Fig.8.6: Marketing and Promotion

- **Example:** Slack, a business communication tool, created awareness by using content marketing early on. They targeted a niche audience of tech-savvy companies that could benefit from their real-time messaging service. They ran targeted ad campaigns on sites like TechCrunch and offered a freemium model, which encouraged users to sign up and experience the product.
- **Approach:** Start with digital marketing channels that resonate with your targeted audience. For example, if you're launching a food delivery app, focus on local SEO, social media ads targeting food lovers, and partnerships with food bloggers to drive interest. Create engaging content (e.g., how-to guides, blog posts, and user stories) to generate organic interest.

5. Feedback Channels

Fig.8.7: Feedback Channels

- **Strategy:** Set up clear channels for users to provide feedback within the product. Encourage users to report issues, suggest improvements, and share their experiences.
- **Example:** Tesla has a built-in feedback system in their cars where users can report bugs, suggest new features, and provide feedback directly to the company. They also engage with users through social media and forums to understand their needs and improve their products based on this input.
- **Approach:** You can integrate an in-app feedback form or provide an email address for users to contact your support team. It's important to not just collect feedback but to act on it by updating the product and informing users of the improvements based on their suggestions. This builds trust and helps in refining the MVP.

6. Monitoring and Analytics

Fig.8.8: Monitoring and Analytics

- **Strategy:** Implement robust analytics tools to track user behavior, engagement, and key performance metrics. This data will help you make informed decisions and measure the MVP's success.
- **Example:** Spotify uses advanced analytics to track user behavior, including the types of music played, skips, and playlist creation. This allows them to understand user preferences and personalize

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

recommendations, which directly affects engagement and retention.

- **Approach:** Set up analytics tools like Google Analytics, Mixpanel, or Amplitude to track how users interact with your MVP. Measure important metrics like user retention, conversion rates, feature usage, and session duration. If your product is an app, track install-to-sign-up conversion rates and feature adoption to understand how users are engaging with the core functionality of the MVP.

7. Customer Support

- **Strategy:** Provide responsive customer support to address user questions, issues, and concerns promptly. A positive support experience can lead to user retention and referrals.

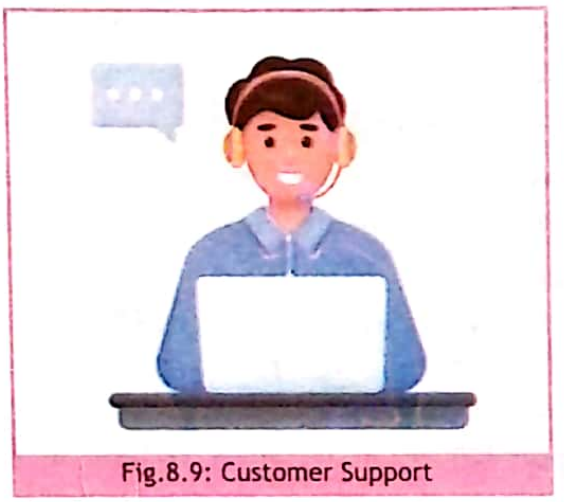


Fig.8.9: Customer Support

- **Example:** Zappos, the online shoe retailer, is known for exceptional customer service. They offer free returns and 24/7 customer support, ensuring that users feel supported throughout their buying journey. This has built a loyal customer base that values the brand's dedication to service.
- **Approach:** Offer multiple support channels, such as live chat, email, and a comprehensive FAQ section. Additionally, creating a community forum or leveraging user guides and video tutorials can help users find answers to common issues without needing to contact support. Addressing problems quickly and positively builds a strong reputation, especially during the MVP stage when bugs or limitations are inevitable.

Summary of Approach:

Launching an MVP requires a well-thought-out strategy to ensure that you're focusing on the right audience, iterating based on feedback, and delivering a strong first impression. By following these strategies and examples, you can reduce the risks associated with a new product, build a loyal user base, and ensure that your MVP can scale effectively once it's ready for a broader release.

8.1.3 Collecting User Feedback and Data

Collecting user feedback and data is crucial during the MVP phase to drive continuous improvement:

Feedback Mechanisms: Implement various feedback mechanisms within the MVP, such as in-app surveys, feedback forms, or direct contact options. Make it easy for users to share their thoughts.

User Interviews: Conduct one-on-one interviews or focus groups with users to gain deeper insights into their experiences and pain points.

Analytics Tools: Use analytics tools to track user behavior and gather quantitative data. Monitor user engagement, conversion rates, and other relevant metrics.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Feedback Analysis: Regularly review and analyze the feedback and data collected. Look for patterns and trends that can inform feature enhancements and optimizations.

Iterative Development: Based on user feedback and data analysis, prioritize and implement changes and new features in subsequent iterations of the MVP.

Engagement Metrics: Pay attention to user retention rates and user engagement metrics. These indicators can reveal the MVP's stickiness and user satisfaction.

Feature Validation: Use data and user feedback to validate or pivot on certain features or aspects of the product. Ensure that changes align with user needs and preferences.

The MVP launch is just the beginning of your product's journey. By actively collecting user feedback and data, you can make informed decisions, continuously improve the product, and work towards achieving your long-term product goals. This iterative approach is key to creating a successful and user-centric product.

Fig.8.10: Feedback Mechanisms

8.2 Iterative Development

8.2.1 The Concept of Continuous Improvement

Continuous improvement, often referred to as iterative development, is a fundamental principle in the product development process. It involves a cyclical approach to refining and enhancing a product or service over time. The core idea is to never stop looking for ways to make your product better, more efficient, and more aligned with user needs and market trends.

Key aspects of continuous improvement include:

Feedback-Driven: Continuous improvement relies heavily on feedback from users, stakeholders, and data analysis. It's about actively seeking and listening to input and insights that can inform changes and refinements.

Iterative Cycles: Development occurs in iterative cycles or sprints, with each cycle focused on specific improvements or enhancements. These cycles are typically short and result in frequent updates or releases.

Flexibility: Teams must be adaptable and open to change. The ability to pivot or adjust course based on feedback and new information is a hallmark of continuous improvement.

Prioritization: Not all improvements are equal. Prioritization is essential to focus efforts on changes that have the most significant impact on user satisfaction, usability, or business goals.

Data-Driven Decision-Making: Decisions regarding improvements should be supported by data and evidence, rather than assumptions or opinions.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

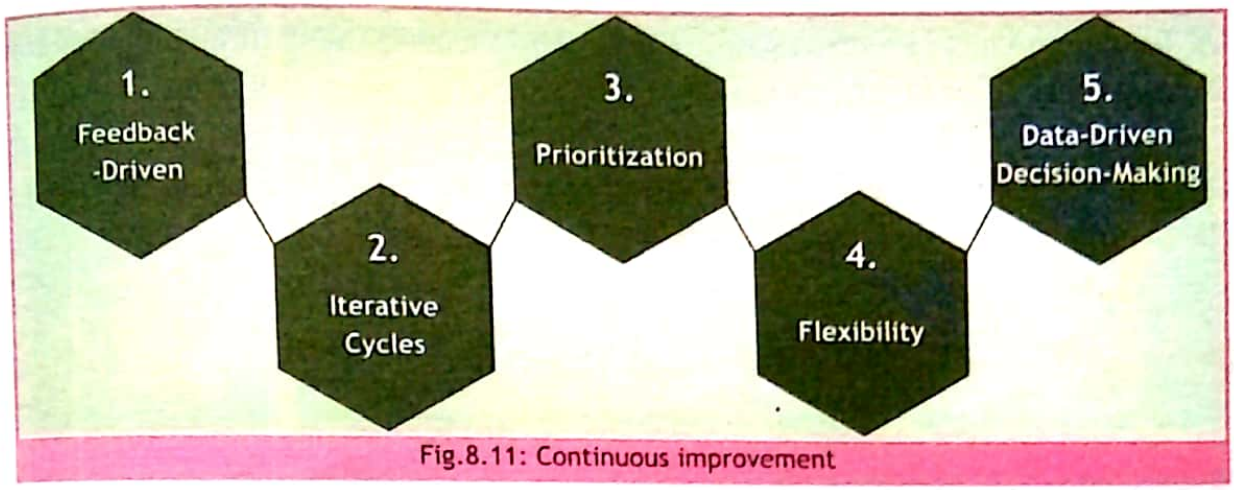


Fig.8.11: Continuous improvement

8.2.2 Using Feedback to Refine and Enhance the Product or Service

Feedback plays a central role in the iterative development process. Here's how to effectively use feedback to refine and enhance your product or service:

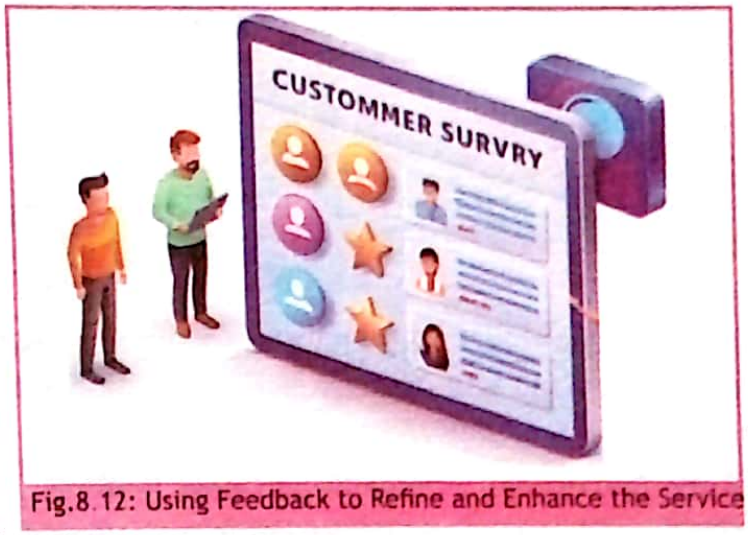


Fig.8.12: Using Feedback to Refine and Enhance the Service

Gathering Feedback: Actively seek feedback from users through various channels, including in-app feedback forms, surveys, customer support interactions, and user testing sessions.

Data Analysis: Analyze user behavior and engagement data to identify patterns, trends, and areas where users may be experiencing difficulties or drop-offs.

Prioritization: Prioritize feedback and improvement ideas based on their potential impact and alignment with your product goals. Not all feedback should result in immediate action; some may be deferred for future iterations.

Clear Objectives: Define clear objectives for each iteration. What specific improvements or changes are you aiming to achieve, and how will you measure success?

Cross-Functional Collaboration: Collaboration among multidisciplinary teams (designers, developers, marketers, etc.) is crucial to implement feedback effectively. Different teams can contribute their expertise to address various aspects of the product.

User Testing: Continuously conduct user testing with real users to validate proposed improvements and gather fresh feedback on the changes made.

Communication: Keep stakeholders informed about the progress and outcomes of each iteration. Transparent communication helps maintain alignment and support for the continuous improvement process.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

8.2.3 Scaling and Expanding Based on MVP Success

As your MVP (Minimum Viable Product) gains traction and success, the natural progression is to scale and expand. Here's how to do it effectively:

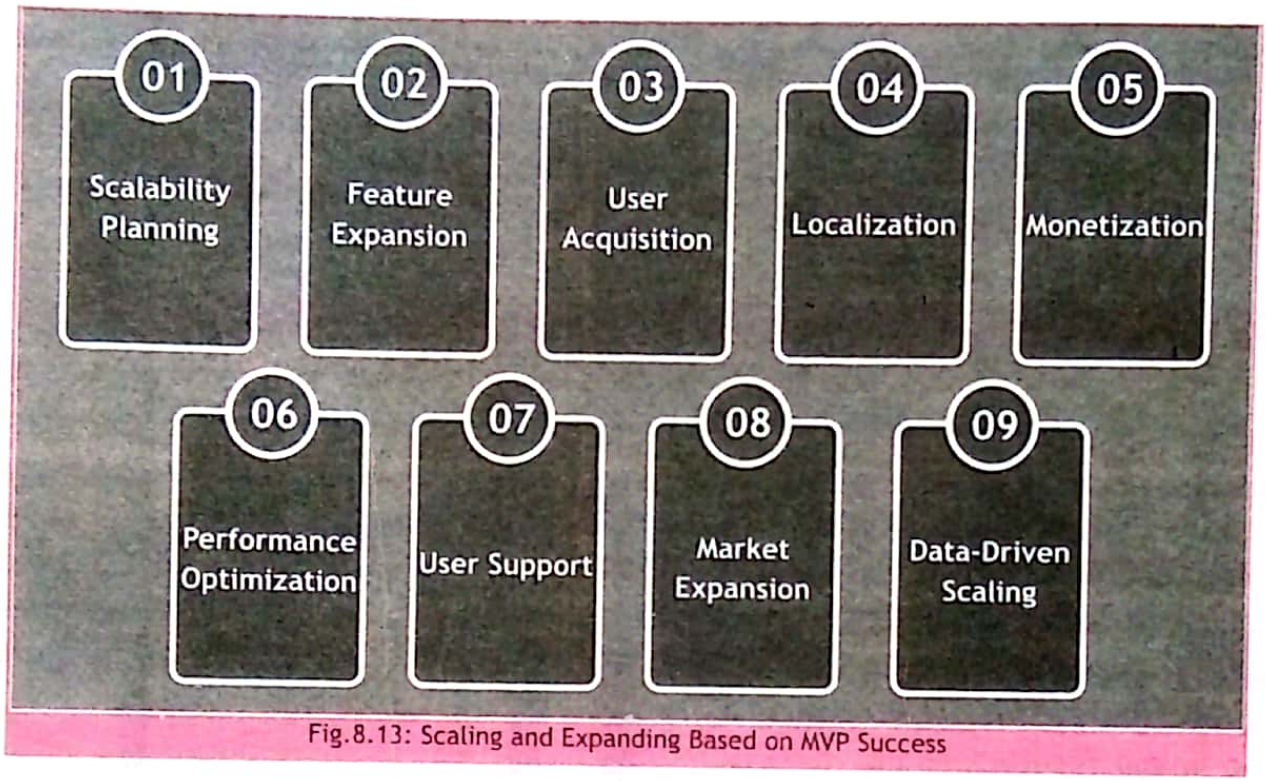


Fig.8.13: Scaling and Expanding Based on MVP Success

Scalability Planning: Ensure that the architecture and infrastructure of your product can handle increased demand and usage as you scale. Be prepared to invest in server capacity, security measures, and scalability features.

Feature Expansion: Gradually add new features and functionalities based on user needs, feedback, and market trends. However, maintain a focus on simplicity and user-centric design, avoiding feature bloat.

User Acquisition: Develop strategies to acquire more users. This may involve marketing efforts, partnerships, or referral programs. Consider user acquisition costs and strategies for retaining new users.

Localization: If applicable, consider localization for different regions or languages to expand your user base globally.

Monetization: Explore different monetization models such as subscriptions, advertising, or in-app purchases. The choice of monetization should align with your user base and product niche.

Performance Optimization: Continuously monitor and optimize the performance of your product, ensuring that it remains fast, reliable, and user-friendly even as it scales.

User Support: Be prepared to scale your customer support and user assistance capabilities to handle increased user inquiries and issues.

Market Expansion: If your MVP has succeeded in a specific market or niche, consider expanding to related markets or niches with similar needs and preferences.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Data-Driven Scaling: Make scaling decisions based on data and user behavior. Avoid premature scaling that can strain resources unnecessarily.

In summary, continuous improvement through iterative development is a core principle in product development. It involves using feedback and data to refine and enhance your product, and as your MVP succeeds, you can strategically scale and expand to reach a broader audience and achieve long-term success.

8.2.4 Evaluation Criteria for Prototypes and MVPs

When evaluating prototypes and Minimum Viable Products (MVPs), it's essential to consider specific criteria that align with the goals of the project. Here are some evaluation criteria:

Alignment with User Needs: Assess how well the prototype or MVP addresses the identified problem or need of the target audience. Does it effectively solve the problem?

Usability: Evaluate the user interface, navigation, and overall user experience. Is the product easy to use, and does it provide a smooth interaction flow?

Functionality: Assess whether the product's core features and functionalities work as intended. Are there any technical issues or bugs?

Scalability: Consider the potential for scalability and future growth. Is the architecture and infrastructure designed to handle increased demand?

Market Fit: Evaluate whether the product aligns with market demands and trends. Does it offer a unique value proposition or competitive advantage?

Feedback Incorporation: Assess how well user feedback has been incorporated into the prototype or MVP. Has feedback-driven improvement been evident?

Iterative Development: Consider the evolution of the product over time. Has it gone through multiple iterations to enhance its features and user experience?

Performance Metrics: Use relevant metrics to measure the success of the MVP. For example, track user engagement, conversion rates, or user satisfaction scores.

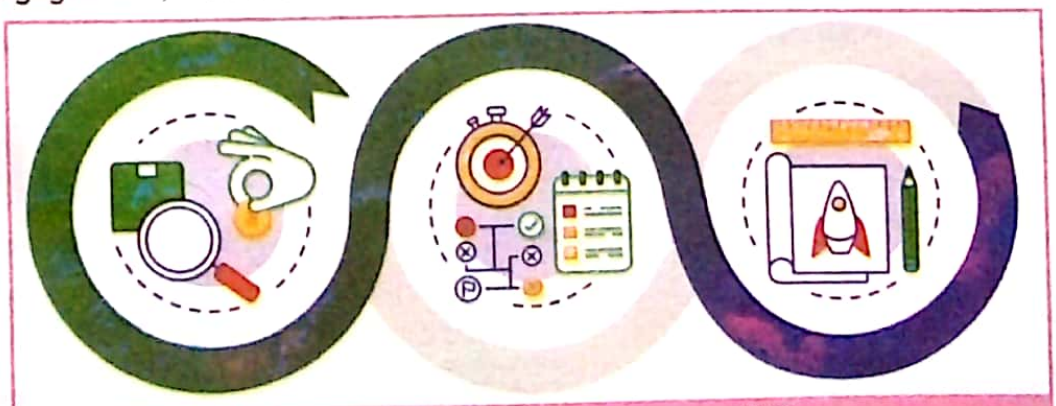


Fig.8.14: Evaluation Criteria for Prototypes and MVPs

8.2.5 Providing Constructive Feedback

Providing constructive feedback is essential for guiding students' learning and improvement. Here are tips for offering effective feedback:

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Specificity: Be specific about what the student did well and what needs improvement. Vague feedback is less helpful.

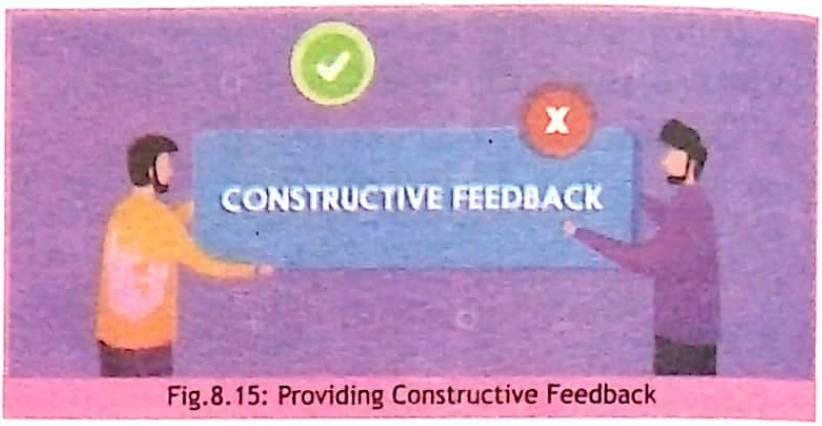


Fig.8.15: Providing Constructive Feedback

Balance: Offer a balance of positive feedback and areas for improvement. Acknowledge the student's strengths before addressing weaknesses.

Focus on the Work, Not the Student: Frame feedback in terms of the project or assignment rather than making it personal. For example, say, "The report lacks sufficient data analysis," instead of "You didn't analyze the data well."

Use the "Feedback Sandwich": Start with positive feedback, provide constructive criticism, and end with encouraging remarks or suggestions for improvement.

Suggest Actionable Steps: Offer specific suggestions or steps the student can take to address the identified areas for improvement.

Encourage Self-Reflection: Prompt students to reflect on their work and what they've learned from the feedback. This encourages metacognition.

Timeliness: Provide feedback promptly, ideally soon after the completion of assignments or assessments, so it's still relevant to the student.

Open Dialogue: Encourage students to ask questions or seek clarification about the feedback. Foster a dialogue to ensure they understand and can act on the feedback.

Remember that effective feedback is a two-way communication process, and it should ultimately help students grow and develop their skills in product development.

8.3 Conclusion and Future Directions

8.3.1 Recap of Key Concepts and Skills Learned

In this course on product development, you've gained valuable knowledge and skills essential for creating successful products and services. Let's recap some of the key concepts and skills you've learned:

Product Development Process: You've explored the step-by-step process of product development, from ideation and prototyping to launching Minimum Viable Products (MVPs) and iterative development.

Prototyping: You've learned the importance of prototyping and how to create low-fidelity and high-fidelity prototypes to validate your product ideas.

User Testing: You've discovered the significance of user testing in refining your products, gathering feedback, and ensuring a user-centric approach.

Iterative Development: You've embraced the concept of continuous improvement, where

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

feedback and data drive the enhancement of your products over time.

Launching MVPs: You've explored strategies for launching MVPs and scaling products based on their success, all while maintaining a user-centric focus.

Assessment and Evaluation: You've understood how to assess student progress and how to evaluate prototypes and MVPs effectively.

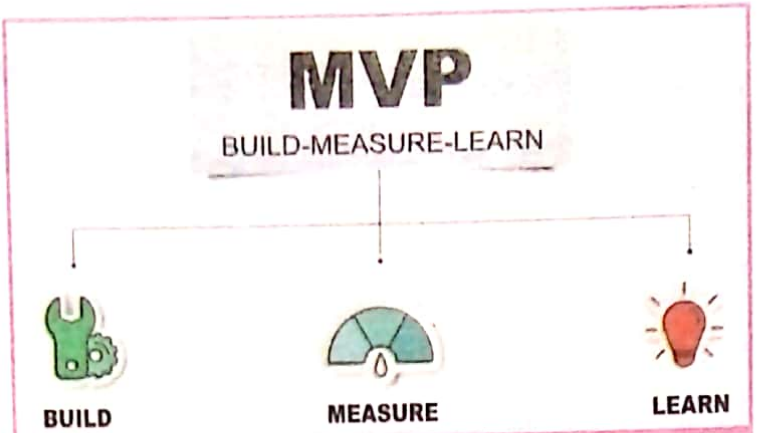


Fig.8.16: Recap of Key Concepts and Skills Learned MVP

8.3.2 Encouragement for Students to Apply These Principles in Real-Life Projects

As you move forward in your academic and professional journey, remember that the principles and skills you've learned in product development are highly transferable and valuable in real-life projects. Here's some encouragement:

Hands-On Experience: Seek opportunities to apply what you've learned in real-world projects or internships. Practical experience will deepen your understanding and expertise.

Entrepreneurship: If you're inclined towards entrepreneurship, consider starting your own venture. Apply the principles of product development to bring your ideas to life.

Innovation: Whether you work for a startup or a well-established company, innovation is key. Use your knowledge of product development to contribute fresh ideas and improvements to existing products or services.

Problem Solving: Product development skills are problem-solving skills. Apply them to tackle complex challenges in various domains, from technology and healthcare to education and beyond.

Collaboration: Collaboration is often essential in product development. Work effectively in interdisciplinary teams, leveraging each member's expertise to create successful products.

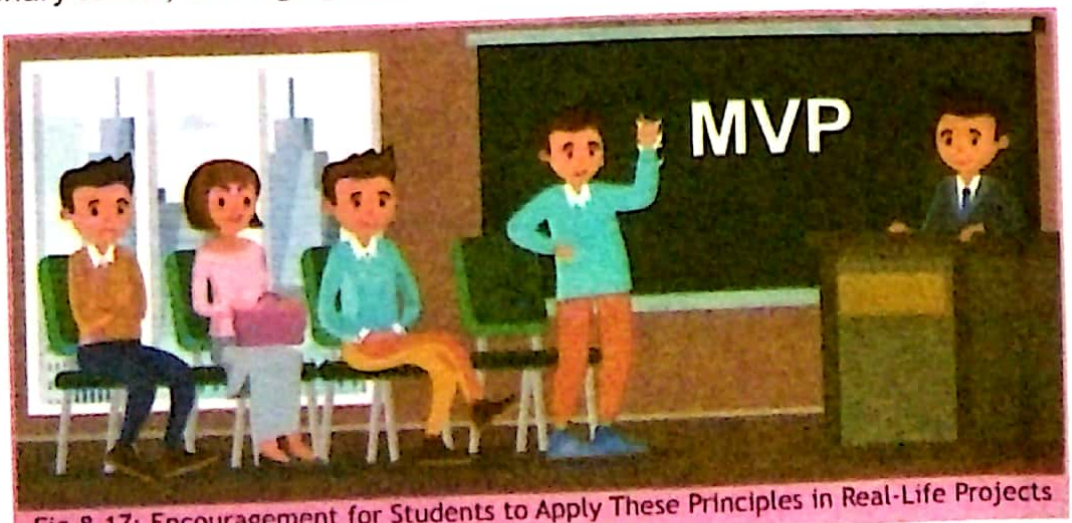


Fig.8.17: Encouragement for Students to Apply These Principles in Real-Life Projects

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

8.3.3 Mention of Potential Career Paths Related to Product Development

Product development skills open doors to a variety of exciting career paths. Here are some potential directions you might consider:

Product Manager: As a product manager, you'll oversee the development of a product from concept to launch, making strategic decisions and collaborating with cross-functional teams.

UX/UI Designer: Focus on user experience and user interface design, creating visually appealing and user-friendly products.

Software Developer/Engineer: Build and maintain software products, coding the features and functionalities envisioned in the product development process.

Entrepreneur: Start your own company and use product development principles to bring your innovative ideas to the market.

Business Analyst: Analyze market trends, user data, and feedback to make data-driven decisions in product development.

Project Manager: Manage the execution of product development projects, ensuring they are completed on time, within budget, and meet quality standards.

Innovation Consultant: Help organizations identify opportunities for innovation and guide them through the product development process.

Marketing Manager: Develop and execute marketing strategies for products and services, understanding the product's unique value proposition.

The field of product development is dynamic and offers a range of opportunities for those who are passionate about creating impactful and user-centered solutions. Whether you choose a career in technology, healthcare, entertainment, or any other industry, the principles you've learned here will serve as a solid foundation for success. Embrace the journey ahead with enthusiasm, and continue to learn and grow in the exciting world of product development.

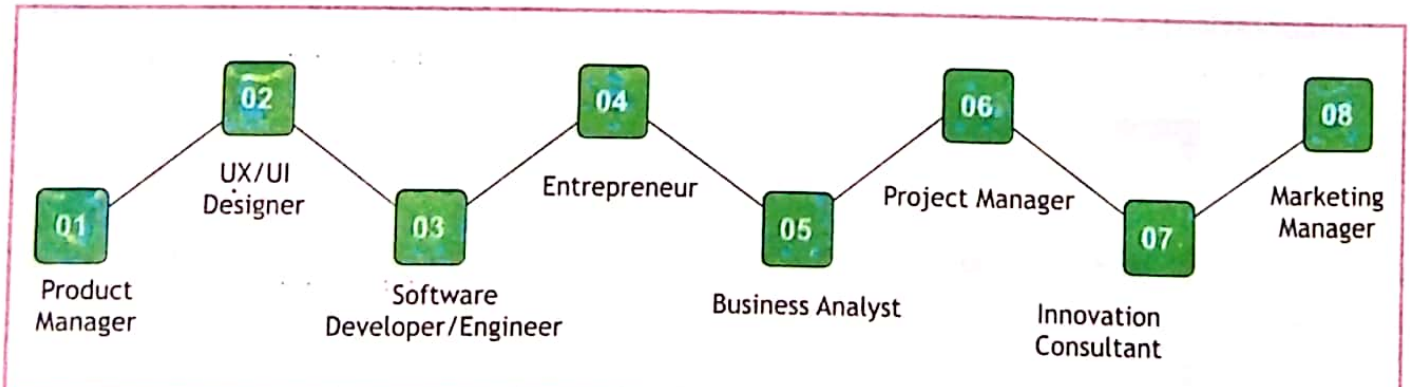


Fig.8.18: Potential Career Paths Related to Product Development

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Case Study

Developing an MVP for a Sustainable Packaging Solution

Background:

A group of students is tasked with developing a Minimum Viable Product (MVP) for a new business that aims to tackle the growing issue of plastic waste in packaging. The students are required to identify a problem related to sustainability, brainstorm possible solutions, create an MVP, and present their business solution. Their goal is to launch a product or service that reduces the environmental impact of packaging while also meeting the needs of consumers and businesses.

Step 1: Identifying the Problem

Problem Identification:

The team conducts research on the environmental impact of plastic packaging and identifies the following issues:

- High Plastic Waste: Over 300 million tons of plastic are produced every year, with much of it used in packaging that ends up in landfills and oceans.
- Consumer Awareness: While consumers are increasingly aware of the need for sustainability, many still struggle to find accessible, affordable, and eco-friendly packaging alternatives.
- Business Pressure: Companies face pressure to meet sustainability goals, but the transition to green packaging solutions is often expensive or logistically difficult.

Customer Research:

To validate their findings, the team surveys local businesses, consumers, and environmental experts. They find that consumers prefer sustainable products, but are often unwilling to pay a premium for eco-friendly packaging. Businesses express frustration about the high cost of transitioning to greener alternatives.

Step 2: Creating Business Solutions

Brainstorming and Ideation:

The team holds a brainstorming session to generate solutions. A few ideas they consider include:

- Compostable Packaging: Packaging that breaks down naturally, such as plant-based materials like cornstarch or mushrooms.
- Reusable Packaging: Packaging that can be returned and reused, such as glass or metal containers.
- Eco-friendly Shipping Materials: Alternative shipping materials that are both sustainable and protective for products, such as biodegradable peanuts or cardboard made from recycled materials.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

They decide to focus on **compostable packaging** made from plant-based materials, which they believe will be most accessible to consumers and businesses alike.

Feasibility Study:

The team conducts a feasibility study and determines that while plant-based materials are an effective alternative to plastic, there are challenges related to cost, scalability, and consumer acceptance. However, they believe they can mitigate these challenges by offering the product as a more affordable, bulk-purchase solution for small and medium-sized businesses, who are often the least able to afford expensive eco-friendly packaging.

Value Proposition:

Their value proposition is: "Affordable, compostable packaging solutions for small businesses that help reduce environmental impact and meet sustainability goals without breaking the bank."

Step 3: Building the Minimum Viable Product (MVP)

Core Features:

The students focus on developing an MVP that includes:

- Compostable Packaging: Simple, single-use packaging made from plant-based materials, such as cornstarch or sugarcane pulp.
- Custom Branding: The ability for businesses to print their logos and designs on the packaging to enhance brand recognition.
- Bulk Purchase Option: The option for businesses to order in bulk at a lower price point to make the solution more affordable.

Prototyping:

The team creates wireframes and mock-ups of the packaging design and how it will look when printed with branding. They also prototype a basic website for businesses to place orders.

MVP Development:

The students use a low-code platform to quickly set up an online store where businesses can place orders for the compostable packaging. They use a third-party supplier to create a small batch of the packaging to test with local businesses.

Iterative Feedback:

After launching the MVP, the students collect feedback from local small business owners. They learn that some businesses want packaging with more varied sizes and shapes, while others are interested in ensuring the material meets specific durability standards. They decide to iterate on the MVP by offering a few different sizes and improving the strength of the material.

Step 4: Presenting the Business Solution

Elevator Pitch:

The team crafts an elevator pitch: "Our sustainable packaging solution helps small businesses

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

reduce their carbon footprint by offering compostable packaging that's both eco-friendly and cost-effective. By switching to our affordable, plant-based packaging, businesses can meet sustainability goals and show consumers they care about the planet."

Business Model Canvas:

They use a Business Model Canvas to outline their plan:

- Key Partners: Suppliers of plant-based materials, local delivery companies.
- Key Activities: Product sourcing, packaging design, marketing, and customer support.
- Key Resources: Website platform, design tools, and supplier relationships.
- Value Proposition: Affordable and eco-friendly compostable packaging for small businesses.
- Customer Segments: Small to medium-sized businesses in the food, beauty, and retail industries.
- Channels: Online store, social media, and partnerships with environmental organizations.
- Revenue Streams: Direct sales, subscription model for regular deliveries.

Pitch Deck:

The team creates a pitch deck that includes:

1. Problem: The environmental impact of plastic packaging.
2. Solution (MVP): Affordable compostable packaging for small businesses.
3. Market Opportunity: Growing demand for sustainable products and packaging.
4. Business Model: Direct-to-business sales with a focus on bulk orders.
5. Go-to-Market Strategy: Marketing through social media, partnerships with eco-friendly influencers, and offering free samples to local businesses.
6. Financials/Projections: Expected costs, revenue from bulk orders, and long-term scalability.

Storytelling:

The team tells a compelling story about how their solution came about by combining their passion for the environment and their desire to help small businesses succeed. They emphasize the global need for sustainable solutions and how small actions, like switching packaging, can have a big impact.

Step 5: Practical Experience

Hands-On Learning:

Students learn by creating the MVP and testing it in the real world with actual businesses. They face challenges like manufacturing delays and unexpected costs but also gain insight into customer preferences and the need for continuous product iteration.

Peer Feedback:

The students present their pitch to their peers, receiving feedback about potential risks, such

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

as competition from larger packaging companies. They refine their presentation and value proposition based on the feedback they receive.

Assessment of Learning:

The students are evaluated on:

- Problem Identification: How well they identified a pressing environmental issue and understood the needs of their target customers.
- Business Solution: The strength of their solution in addressing the problem, its feasibility, and its value proposition.
- MVP Development: The effectiveness of their MVP in solving the problem while keeping it simple and scalable.
- Presentation: The clarity and persuasiveness of their pitch, the completeness of their business model, and the professionalism of their final presentation.

Conclusion:

By the end of the project, the students have not only developed a viable business solution but also learned important lessons about product development, market research, and the iterative process of refining a product based on user feedback. They now have the skills to identify problems, create innovative solutions, build an MVP, and present a business idea effectively.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Summary

- **Lean Start-Up:** Process of rapidly developing simple prototypes to test key assumptions with real customers.
- **Minimum Viable Product (MVP):** Simplest version of a product that enables sustainable business creation.
- **Dropbox:** A platform to share photos, documents, and videos across devices, using a video for initial testing.
- **User Testing:** Ensures the prototype evolves into a final product that meets user needs and enhances experience.
- **Feature Prioritization:** Means to Identify core features necessary to address primary problems or needs of target audience.
- **MPV Scalability:** Design the MVP for future scalability in architecture and technology.
- **User Onboarding:** Ensure users can easily understand and use the product effectively.
- **Targeted Audience:** Identify specific segments to focus initial marketing efforts.
- **Soft Launch:** Gather feedback from a smaller user base before broader release.
- **Beta Testing:** Invite select users to uncover potential issues and generate early reviews.
- **Marketing and Promotion:** Develop a plan to create awareness and drive user acquisition.
- **Monitoring and Analytics:** Implement tools to track user behavior and measure success.
- **Customer Support:** Provide responsive support to enhance user retention and referrals.
- **Feedback Mechanisms:** Use surveys and direct contact options for user feedback.
- **User Interviews:** Conduct interviews for deeper insights into experiences and pain points.
- **Analytics Tools:** Monitor user engagement and relevant metrics for data-driven decisions.
- **Engagement Metrics:** Track user retention rates and satisfaction indicators.
- **Continuous Improvement:** Refine products iteratively based on user feedback and data analysis.
- **Iterative Cycles:** Develop in short cycles for frequent updates.
- **Data-Driven Decision-Making:** Support improvements with evidence rather than assumptions.
- **User Acquisition:** Develop strategies for attracting and retaining users.
- **Performance Optimization:** Monitor product performance to maintain user-friendliness.
- **Market Expansion:** Consider related markets for broader reach.
- **Principles Application:** Apply learned principles in real-world projects and entrepreneurial ventures.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. What is the primary goal of developing a Minimum Viable Product (MVP)?
   a. To include as many features as possible
   b. To create the most advanced version of the product
   c. To address the core problem or need of the target audience with the simplest version
   d. To achieve the highest possible user engagement immediately
2. Which strategy is NOT typically recommended when launching an MVP?
   a. Conducting a soft launch to gather initial feedback
   b. Expanding the product's features significantly before launch
   c. Using targeted marketing to reach the most relevant audience
   d. Inviting a select group of users for beta testing
3. What role does user feedback play in iterative development?
   a. It is used to maintain the product's original design without changes
   b. It helps in making decisions about scaling the product
   c. It drives continuous improvement and informs necessary changes
   d. It is collected but rarely used for making product enhancements
4. Which of the following is NOT a key aspect of continuous improvement in product development?
   a. Feedback-driven adjustments
   b. Long-term, static product design
   c. Iterative cycles or sprints
   d. Data-driven decision-making
5. What should be considered when scaling an MVP to a broader audience?
   a. Maintaining the product's simplicity and avoiding feature bloat
   b. Avoiding any changes to the MVP based on user feedback
   c. Investing heavily in new features without assessing user needs
   d. Ignoring performance optimization and focusing solely on marketing
6. What is a crucial method for collecting user feedback during the MVP phase?
   a. Ignoring user feedback and focusing on internal opinions
   b. Implementing feedback mechanisms like in-app surveys and direct contact options
   c. Conducting feedback sessions only after a year of product launch
   d. Using only external market research without user input
7. When assessing prototypes and MVPs, what criterion is used to evaluate how well the product addresses user needs?
   a. The complexity of the product's features
   b. The alignment with user needs and effectiveness in solving the problem
   c. The number of features included in the prototype
   d. The frequency of updates and releases

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

8. What is one of the primary benefits of conducting user testing for a prototype or MVP?
   a. It delays the product launch to add more features
   b. It ensures that the product remains static and unchanged
   c. It helps validate assumptions and gather feedback to refine the product
   d. It avoids the need for future iterations and improvements
9. In the context of product development, what is meant by "scalability"?
   a. The ability to add new features without affecting the existing ones
   b. The capability of the product's infrastructure to handle increased demand
   c. The process of reducing the product's features to simplify it
   d. The speed at which the product can be developed initially
10. What is the purpose of a soft launch in the MVP launch strategy?
   a. To make the product available to everyone immediately
   b. To gather feedback from a smaller user base before a broader release
   c. To launch the product with all features fully developed
   d. To avoid collecting user feedback altogether
11. Which of the following best illustrates the use of targeted marketing for an MVP?
   a. Launching the product worldwide without prior research
   b. Advertising to all age groups equally
   c. Focusing on new mothers for a postpartum fitness app
   d. Offering free subscriptions to random users
12. What is the role of analytics in MVP development?
   a. To create advertisements for the product
   b. To enhance product design without user input
   c. To track user behavior and make informed improvements
   d. To reduce the number of users on the platform

Give short answers to the following Short Response Questions (SRQs).

1. What is the primary purpose of a Minimum Viable Product (MVP) in the lean start-up process?
2. Name two key strategies for launching an MVP effectively.
3. How does user feedback contribute to the iterative development process?
4. What is one important consideration when scaling an MVP to a larger audience?
5. Describe one method for collecting user feedback during the MVP phase.
6. Why is simplicity emphasized when developing an MVP?
7. How can customer feedback be effectively collected and used during the MVP phase?

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Content Authors

Mohammad Sajjad Heder has done Master (MS) in Computer Science from George Mason University, Virginia, USA and B.E. Mechanical Engineering from Kim Chaek University of Technology, Korea. He has 25 years' experience of teaching the subject of Computer Science to students of SSC and HSSC. He is a highly experienced author who has written many textbooks of Computer Science.

Lubna Kousar Janjua is an educationist and IT professional, with more than 20 years of experience. She received her MCS degree from IIUI and completed MS coursework from NUST with a specialization in Intelligent Information System. She worked as a software engineer for a multinational company and is currently serving as a Computer Science professor.

Shah Islam Khan, a distinguished author renowned for his groundbreaking 'IQ Series' books. His academic journey boasts a breadth of qualifications, from an MPhil in Higher Education to a range of degrees including BS(CS), DIT, MEd, and BEd. His diverse expertise enriches his literary contributions, promising readers a depth of knowledge and insight unparalleled in his field. Prof Shah Islam has been working in OPF BOYS COLLEGE ISLAMABAD for the last twenty years.

Mohammad Khalid, a luminary in the field of Computer Science and Assistant Professor at OPF Boys College. Professor Khalid's rich pedigree, boasting an M.Sc. in Computer Science and B.Ed. qualifications, infuses every page of these meticulously crafted textbooks for Grades IX to XII. Collaborating with the prestigious National Book Foundation (NBF), Professor Khalid's work epitomizes innovation, excellence, and pedagogical mastery. Dive into a realm of unparalleled insight, where theoretical depth converges with practical application, propelling students towards digital fluency and technological prowess.

Dr. Saleem Iqbal received both his BS and MS in Computer Science from the COMSATS, Pakistan and PhD from UTM, Malaysia in 2015. He has a diversified experience of over 18 years comprising academic, development, research and administrative. Currently, he is an Associate Professor and Chairman of Computer Science Department at Allama Iqbal Open University. Previously he was with PMAS-Arid Agriculture University Rawalpindi and COMSATS, Islamabad.

Dr. Faraz Ahsan is an Educational and IT Specialist with more than 20 years of experience of both academia and industry. He worked as a Software Engineer and Network Administrator in the industry. Later, he completed his PhD from COMSATS Institute of Information Technology and specialized in the field of Network Security. He is currently associated as Consultant with a USA based firm.

Dr. Mian Hamid Hassan is an Educational Specialist with more than 25 years of diversified experience in the field of Education specially teachers training and education. He did Masters in Technology Education from University of the Punjab with distinction and was awarded University Grants Commission's research scholarship for Ph.D. studies. Dr. Hassan is working as an Associate Professor at Federal College of Education and is involved in Teachers Training (In/ Pre - Service) especially in the Field of Technical, Elementary, Secondary and Science education.

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner

Approved by National Curriculum Council, Secretariat
Ministry of Federal Education & Professional Training
vide letter No. F.No.1-1(2)TB/CS/NBF/NCC, Dated: 4th June, 2025

پاک سَر زمین شاد باد! کشورِ حسین شاد باد!
تو نِشانِ عزمِ عالی شان ارضِ پاکستان
مرکزِ یقین شاد باد!

پاک سر زمین کا نِظام قُوّتِ اُخوّتِ عوام
قوم ، مُلک ، سلطنت پائندہ تابندہ باد!
شاد باد منزلِ مُراد!

پرچَمِ سِتارہ و ہِلال رہبرِ ترقی و کمال
ترجُمانِ ماضی، شانِ حال جانِ اِستقبال
سایۂ خُدائے ذُوالجلال!

National Book Foundation
as
Federal Textbook Board
Islamabad

Downloaded from Atta Shad College Digital Library
Scanned by CamScanner